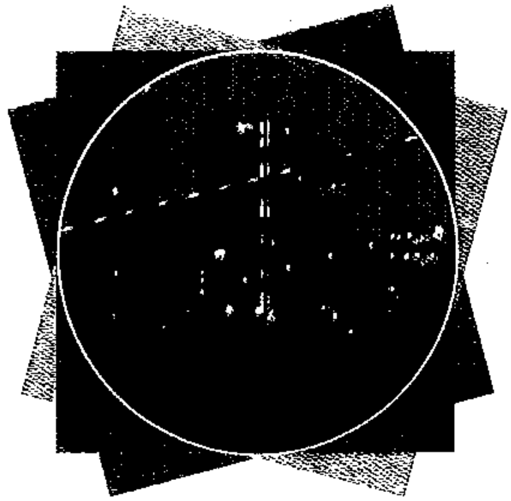

SPECIAL SECTION
Industrial Process Control

Nonlinear Model Predictive Control Using Neural Networks

By Stephen Piché, Bijan Sayyar-Rodsari, Doug Johnson, and Mark Gerules

Model predictive control (MPC), a control algorithm that uses an optimizer to solve for the control trajectory over a future time horizon based on a dynamic model of the process, has become a standard control technique in the process industries over the past two decades. MPC is commonly used for constrained multiple-input/multiple-output (MIMO) control problems which are often encountered in the process industries. MPC has been used in over 2,000 industrial applications in the refining, petrochemical, chemical, pulp and paper, and food processing industries [1].

At the core of the MPC algorithm is a dynamic model. Until recently, most industrial applications have relied on linear dynamic models. These dynamic models have been developed using empirical data obtained from plant testing. Linear, rather than nonlinear, models have been used because of the difficulty in developing a generic nonlinear model from empirical data and also because of the computational expense involved in using nonlinear models in the MPC formulation.

In this article, a neural-network-based technique for developing nonlinear dynamic models from empirical data for an MPC algorithm is presented. These models can be derived for a wide variety of processes and can also be used efficiently in an MPC framework. The nonlinear MPC-based approach presented here has been successfully implemented in a number of industrial applications in the refining, petrochemical, pulp and paper, power, and food industries. Performance of the controller on a nonlinear industrial process, a polyethylene reactor, and a simulated continuous stirred tank reactor (CSTR) is presented.

Motivation for Nonlinear MPC

MPC was developed in the late 1970s and came into widespread use, particularly in the refining industry, in the 1980s

S. Piché, B. Sayyar-Rodsari (bijan@pav.com), D. Johnson, and M. Gerules are with Pavilion Technologies, 11100 Metric Blvd #700, Austin, TX 78758, U.S.A.

1053-5888/00/$10.00©2000IEEE

[1], [2]. The economic benefit of this approach to control in the process industries has been documented. Several factors have contributed to the widespread use of MPC in the process industries:

- *Multivariate Control:* Industrial processes are typically coupled MIMO systems. The number of inputs and outputs can be large (typically on the order of 10-20) and the coupling can be significant. The MPC approach is well suited for MIMO applications.
- *Constraints:* Constraints on the inputs and outputs of a process due to physical constraints and safety considerations are common in the process industries. An example of a physical constraint is the upper limit of an actuator. An example of a safety constraint is an upper limit on the temperature in an exothermic reactor. These constraints can be directly integrated into the control calculation using MPC.

The nonlinear MPC-based approach presented in this article has been successfully implemented in a number of industrial applications in the refining, petrochemical, pulp and paper, power, and food industries.

- *Sampling Period:* Unlike systems in other industries, such as automotive or aerospace, the open-loop settling time for most industrial processes is typically tens of minutes or hours rather than milliseconds. This relatively slow settling time translates to sampling periods measured in minutes. Because the sampling period is sufficiently long, the complex optimization calculations required to implement MPC can be solved at each sampling period.
- *Commercial Tools:* Commercial tools that facilitate model development and controller implementation have allowed the proliferation of MPC in the process industries. These tools allow the user to build dynamic models based on empirical data, tune the controller for robustness in a simulation environment, and implement the controller on the process in real time. In addition to these tools, service organizations that implement these MPC solutions have proliferated.

Until recently, industrial applications of MPC have relied on linear dynamic models even though most processes are nonlinear. MPC based on linear models is acceptable when the process operates at a single setpoint and the primary use of the controller is the rejection of disturbances. Many chemical processes, however, including polymer reactors, do not operate at a single setpoint. These processes are often required to operate at different setpoints depending on the grade of the product to be produced. Because these processes make transitions over the nonlinearity of the system, linear MPC often results in poor control performance. To properly control these processes, a nonlinear model is needed in the MPC algorithm.

This need for nonlinear models in MPC is well recognized. A number of researchers and commercial companies have developed nonlinear models using a variety of technologies, including first-principle and empirical approaches [3], [4]. Although a variety of approaches have been developed, most have not been practical for wide-scale industrial application. For example, nonlinear models built using first principle techniques are expensive to develop and difficult to maintain. In addition, they are often process and operation specific and, therefore, can only be used for a limited number of applications. First-principle-based nonlinear MPC has had only limited industrial use [1], [3].

Until recently, empirically based nonlinear models had also seen limited industrial use. Lack of commercial tools, modeling techniques that relied on costly plant tests in multiple operating regions, and the computational expense of optimizing complex nonlinear models in a real-time environment resulted in limited use of empirically based nonlinear models.

Over the past five years, we have developed a software tool for building, simulating, and implementing nonlinear MPC-based controllers. This tool has been used on more than 100 industrial applications in the refining, chemical, polymers, power, pulp and paper, and food industries. Recently, our nonlinear MPC was recognized as the most widely used nonlinear MPC controller in the process industries [3].

Our approach to nonlinear modeling for MPC differs from earlier attempts. An empirical nonlinear dynamic model is developed using step test data and historical data. The step test data is used to create a linear dynamic model, and the historical data is used to build a nonlinear steady-state model. The steady-state model is implemented by a neural network. As we will show, a parsimonious nonlinear dynamic model is created by combining the dynamic and steady-state models using an advanced form of gain scheduling. The advanced gain scheduling approach allows the gain to vary over the control horizon of the MPC algorithm. Using a parsimonious model reduces the computational expense associated with computing the MPC solution, thus allowing control of nonlinear MIMO processes with many inputs and outputs. Tools for creating the model and implementing the controller are commercially available, allowing for widespread use in the process industries.

We present details of our nonlinear MPC solution. After providing an overview of MPC, details on the formulation of the nonlinear model are presented. We then describe the model, discuss results from a simulation, and present an industrial application.

Model Predictive Control

MPC is based on solving an optimization problem for the control actions at each sampling interval. Using MPC, an optimizer computes future control actions that minimize the difference between a model of the process and desired performance over a time horizon (typically the time horizon is greater than the open-loop settling time of the process). For example, given a linear model of a process

$$y_t = -a_1 y_{t-1} - a_2 y_{t-2} + b_1 u_{t-1} + b_2 u_{t-2}$$

where u_t represents the input and y_t represents the output of the (single-input/single-output) process, the optimizer is used to minimize an objective function

$$J = \sum_{i=1}^{T} \left(\left(y_{t+i} - \hat{y}_{t+i} \right)^2 + w \left(u_{t+i} - u_{t+i-1} \right)^2 \right) \quad (1)$$

where $\hat{y}_t$ is the desired setpoint for the output, w is a weighting factor that allows user balancing of the two terms, and T is the length of the time horizon. The optimization problem is solved subject to a set of constraints. For example, it is common to place upper and lower bounds on the input as well as bounds on the rate of change of the input

$$U_{\text{upper}} \geq u_{t+i} \geq U_{\text{lower}} \quad \forall 1 \leq i \leq T \quad (2)$$

$$\Delta U_{\text{upper}} \geq u_{t+i} - u_{t+i-1} \geq \Delta U_{\text{lower}} \quad \forall 1 \leq i \leq T \quad (3)$$

where U_{upper} and U_{lower} are the upper and lower input bounds and ΔU_{upper} and ΔU_{lower} are the upper and lower rate-of-change bounds. After the trajectory of future control actions is computed, only the first value in the trajectory is sent as a setpoint to the actuators. The optimization calculation is rerun at each sampling interval using a model that has been updated using feedback.

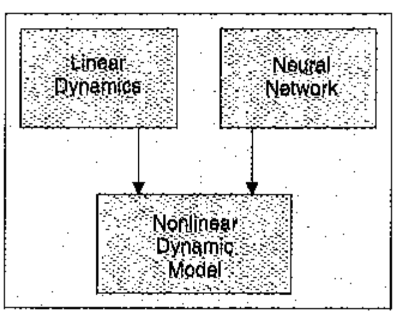

Figure 1. Nonlinear dynamic model: a nonlinear static model represented by a neural network is combined with linear dynamics to create a nonlinear dynamic model.

Active areas of research over the past two decades have been selection of the model, the objective function, the constraints, and the optimizer. Many excellent survey papers on MPC cover these topics [1], [2], [4]. As discussed earlier, we have selected an MIMO nonlinear model (presented in the next section). Although the objective function contains two terms (desired output and input move suppression), the objective function used in our implementation contains 13 terms. Our implementation uses the constraints given in (2) and (3). Because we use nonlinear models, a nonlinear programming technique must be used to solve the optimization problem. We use LS-GRGII, which is a feasible path reduced gradient solver [5].

Components of the Model

The quantity and quality of process data ultimately determine the structure of an empirical model. Here, we show that data available from most processes allows identification of a nonlinear steady-state model and a linear dynamic model. In the next section, we show how these two components are combined to create a nonlinear dynamic model (see Fig. 1).

In the process industries, two types of process data are readily available:

- *Historical Data:* The values of the inputs and outputs of most processes are saved at regular intervals to a database. Furthermore, most processing companies retain historical data associated with their plant for several years.
- *Plant Test Data:* Open-loop plant testing is a well-accepted practice for determining the process dynamics of an MPC application. Most open-loop testing is done by moving a single input and then waiting until the process settles approximately to a steady state. Multiple input moves may improve the quality of the dynamic models.

Most practitioners of MPC have used only plant test data and ignored historical data. Historical data was ignored in the past because it was difficult to extract and preprocess the data and to build models. Historical data was also viewed as not useful because it was collected in closed-loop testing, and therefore process dynamics could not be determined. Thus, by relying only on plant test data, often from a single operating region, only linear dynamic models could be developed.

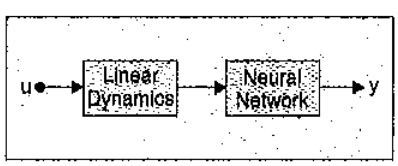

Figure 2. Hammerstein model: the inputs to a static nonlinear model, in this case a neural network, are preprocessed by a linear model representing the dynamics of the process.

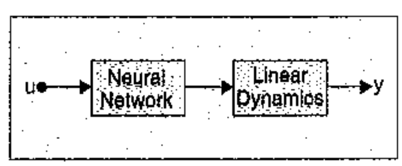

Figure 3. Weiner model: the outputs of a static nonlinear model are postprocessed by a linear model representing the dynamics of the process.

Although historical data may not necessarily be useful for determining the dynamics of the process, it is often useful for determining the nonlinear steady-state relationship between the inputs and outputs. For processes that operate at multiple setpoints, the historical data contains rich information about different steady states of the process. For example, the historical data of a polymer reactor that produces different grades of product contains information on the steady-state inputs and outputs for each of these grades. This historical data can be used to derive an empirically based nonlinear steady-state model.

MPC is based on solving an optimization problem for the control actions at each sampling interval.

The nonlinear steady-state models

$$\mathbf{y}_{ss} = \mathrm{NN}_{ss}(\mathbf{u})$$

are implemented by a feedforward neural network and trained using a variant of the backpropagation algorithm [6]. In developing a neural network model, it is important that the modeler have great flexibility in determining the architecture of the neural network, including the ability to select which inputs affect which outputs. To provide greater flexibility, an algorithm for specifying bounds on the gain (Jacobian) of the model has recently been implemented to guarantee the quality of the model for control applications [7].

The historical data is useful for developing nonlinear steady-state models, while the plant test data is useful for creating linear dynamic models. Using plant test data, second-order models with input time delay are identified

$$y_t = -a_1 y_{t-1} - a_2 y_{t-2} + b_1 u_{t-d-1} + b_2 u_{t-d-2}. \tag{4}$$

Second-order models are used because the amount of plant test data is often limited and the data is often affected by noise and disturbances. Under these circumstances, higher order models do not yield more accurate representations of the process. The parameters of (4) are identified by minimizing the squared error between the model and the plant test data. To prevent a biased estimate of the parameters, the identification problem is solved using an optimizer because of the correlation in the model inputs [8]. Tools for selecting the identification regions and viewing the results are available. Finally, MIMO identification as well as disturbance modeling techniques are allowed for the development of accurate linear dynamic models. As outlined in the next section, combining the nonlinear steady-state model with linear dynamic models derived from the plant test data provides a generic approach to developing nonlinear models.

Combining the Nonlinear Steady-State and Linear Dynamic Models

A variety of techniques are available for combining nonlinear steady-state and linear dynamic models. The linear dynamic models can be used to either preprocess the inputs or postprocess the outputs of the steady-state model as shown in Figs. 2 and 3. These models, referred to as Hammerstein and Weiner models, respectively [8], contain a large number of parameters and are computationally expensive to use in an optimization problem. When based on neural networks, they also extrapolate poorly.

Gain scheduling is often used to combine nonlinear steady-state models and linear dynamic models. As shown in Fig. 4, using a neural network steady-state model, the gain at the current operating point u_i

$$g_i = \frac{\partial y_{ss}}{\partial u}\Big|_{u=u_i}$$

is used to update the gain of the linear dynamic model of (4)

$$\delta y_t = -a_1 \delta y_{t-1} - a_2 \delta y_{t-2} + v_1 \delta u_{t-d-1} + v_2 \delta u_{t-d-2} \tag{5}$$

where

$$v_1 = b_1 g_i \frac{1 + a_1 + a_2}{b_1 + b_2} \tag{6}$$

$$v_2 = b_2 g_i \frac{1 + a_1 + a_2}{b_1 + b_2}. \tag{7}$$

The difference equation is linearized about the point u_i and $y_i = \mathrm{NN}_{ss}(u_i)$, thus $\delta y = y - y_i$ and $\delta u = u - u_i$. To simplify the equations above, a single-input, single-output (SISO) system is used. Gain scheduling results in a parsimonious model that is efficient to use in the MPC optimization problem; however, because this model does not incorporate information about the gain over the entire trajectory, its use results in suboptimal performance of the MPC algorithm.

Our nonlinear modeling approach remedies this problem. By solving a steady-state optimization problem whenever a setpoint change in the output is made, it is possible to compute the final steady-state values of the inputs u_f. This is accomplished by including a steady-state optimizer as part of the MPC controller. Given the final steady-state input values, the gain associated with the final steady state can be computed. For a SISO system, this gain is given by

$$g_f = \frac{\partial y_{ss}}{\partial u}\Big|_{u=u_f}. \tag{8}$$

Using the initial and final gains associated with a setpoint change, the gain structure over the entire trajectory can be approximated. This two-point gain scheduling overcomes the limitations of regular gain scheduling in MPC algorithms.

Combining the initial and final gains with the linear dynamic model as shown in Fig. 5, a quadratic difference equation is derived for the overall nonlinear model

$$\delta y_t = -a_1 \delta y_{t-1} - a_2 \delta y_{t-2} + v_1 \delta u_{t-d-1} + v_2 \delta u_{t-d-2} + w_1 \delta u_{t-d-1}^2 + w_2 \delta u_{t-d-2}^2 \quad (9)$$

where

$$w_1 = b_1 \frac{(1+a_1+a_2)(g_f - g_i)}{(b_1+b_2)(u_f - u_i)} \quad (10)$$

$$w_2 = b_2 \frac{(1+a_1+a_2)(g_f - g_i)}{(b_1+b_2)(u_f - u_i)} \quad (11)$$

and v_1 and v_2 are given by (6) and (7). Use of the gain at the final steady state introduces the last two terms of (9). This model allows the incorporation of gain information over the entire trajectory in the MPC algorithm. The gain of (9) at u_i is g_i, while at u_f it is g_f. Between the two points, the gain is a linear combination of g_i and g_f. For processes with large gain changes, such as polymer reactors, this results in dramatic improvements in the performance of the MPC controller.

An additional benefit of using the model given by (9) is that the user can bound the initial and final gains for each input-output pair and thus determine the amount of nonlinearity used in the model. This gain bounding can be changed either in simulation or during online use. For example, traditionally when implementing a linear MPC on a nonlinear process, the user selects a linear dynamic model that has the maximum gain of the process (if the gain is positive) or the minimum gain (if the gain is negative) for each input-output pair. This gain selection for the model allows the user to implement the MPC controller without having to use large amounts of move suppression to detune the controller. This approach can be emulated using gain bounding. Specifically, a user can set the bounds on the gains to initialize the controller with the maximum or minimum gains of the process, thus starting with a linear controller. After the users become confident in the linear solution, they can introduce nonlinearity into the model by reducing the bounds. After becoming confortable with a limited nonlinear model, the user could eliminate the bounds and use the full nonlinearity of the model. For practitioners who are used to implementing MPC with linear models, using gain bounds allows them to easily transition from linear to nonlinear models. This ability to control the amount of nonlinearity used in the model has been important for ensuring acceptance of this new model in many applications. Finally, gain bounding can be used to guarantee extrapolation performance of the model.

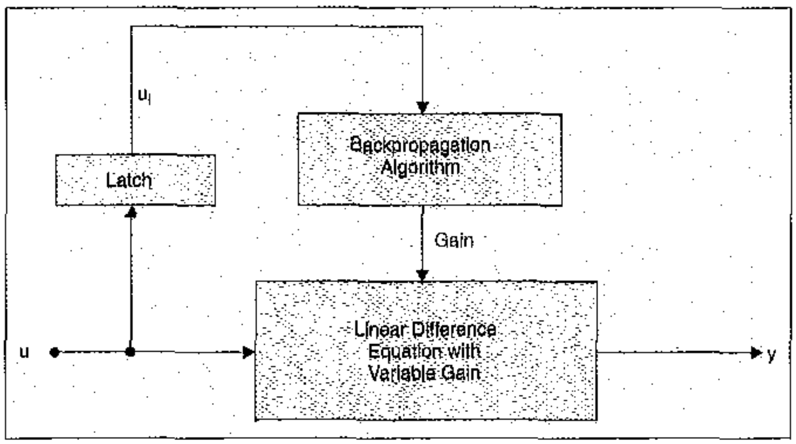

Figure 6. *Neural network gain scheduling model: the gain of a linear difference equation is modified using the gain of a neural network. The backpropagation algorithm is used to compute the gain associated with the current operating point. In most gain scheduling approached, the gain is changed only periodically. Often this is accomplished by latching the input at the beginning of a transition and at the end of a transition.*

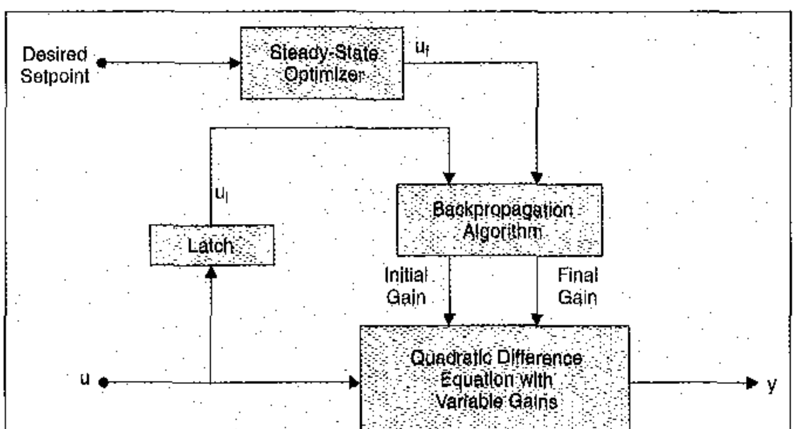

Figure 7. *Two-point gain scheduling model: the gain of a quadratic difference equation is modified using the initial and final gains of a neural network over the trajectory. The final point is determined using a steady-state optimizer and a neural network model. The backpropagation algorithm is used to compute the gain associated with the initial and final points. The initial and final gains are computed at the beginning of a transition and then held constant until the next transition.*

The nonlinear model of (9) fits the criteria that must be satisfied for widespread use of nonlinear models for MPC. The model is based on readily available data and has a parsimonious representation, allowing models with many inputs and outputs to be efficiently used in the optimizer. Furthermore, it addresses the primary nonlinearity found in processes, specifically, the significant changes in gain over the operating region.

CSTR Application

In this section, we examine the performance of the nonlinear MPC controller for a CSTR with a side and consecutive reaction scheme, known as the Van de Vusse reaction [9]. This problem exhibits highly nonlinear dynamics and has been considered a benchmark problem for nonlinear process control algorithms. A detailed description of this problem can be found in [10] and the references therein. It suffices to point out here that in a Van de Vusse reaction, cyclopentenol (B) is produced from cyclopentadiene (A) and the by-products cyclopentanediol (C) and dicyclopentadiene (D) are produced in unwanted side and consecutive reactions. The following set of nonlinear differential equations govern the process dynamics:

$$\frac{dV_R}{dt} = F_{\text{in}} - F_{\text{out}} \tag{12}$$

$$\frac{d(V_R c_A)}{dt} = F_{\text{in}} c_{A\text{in}} - F_{\text{out}} c_A - V_R\left[k_1(T)c_A + k_3(T)c_A^2\right] \tag{13}$$

$$\frac{d(V_R c_B)}{dt} = -F_{\text{out}} c_B + V_R\left[k_1(T)c_A - k_2(T)c_B\right] \tag{14}$$

$$\frac{d(V_R T)}{dt} = F_{\text{in}} T_{\text{in}} - F_{\text{out}} T + \frac{k_W A_R}{\rho C_p}(T_k - T) - \frac{V_R}{\rho C_p}\left[k_1(T)c_A \Delta H_1 + k_2(T)c_B \Delta H_2 + k_3(T)c_A^2 \Delta H_3\right] \tag{15}$$

$$\frac{dT_K}{dt} = \frac{1}{m_K C_{pK}}\left[Q_k + k_W A_R(T - T_k)\right]. \tag{16}$$

Using (12)-(16), a process simulator was developed. For simulation and control, heat removal rate Q_k is an input to the model and a manipulated variable in the controller. Other inputs to the process, inlet temperature T_{in}, inlet concentration of A, $c_{A\text{in}}$, and input volume flow F_{in} are treated as disturbance variables and, therefore, are not manipulated by the controller. It is assumed that a proportional-integral-derivative (PID) loop is used to maintain the tank level at V_R (i.e., $F_{\text{in}} = F_{\text{out}}$). The controlled variable is the concentration of the desired substance, c_B, in the reactor. (To maintain a desired level for c_B in the face of disturbance in T_{in} and $c_{A\text{in}}$, the control input Q_k alone is not enough. Therefore, the input flow F_{in} is also used as a manipulated variable when needed.) Each specific reaction rate k_i is assumed to depend on the temperature via an Arrhenius equation.

A nonlinear model of the process was developed using input-output data from a simulator. As mentioned previously, the nonlinear model is composed of a combination of the nonlinear steady-state model and a linear dynamic model. The performance of this modeling approach is shown in Fig. 6, where the dynamic response for the identified nonlinear dynamic model and the simulated CSTR process are compared. By comparing the predicted versus actual response in Fig. 6, it can be observed that the nonlinear gain structure is captured by the steady-state neural network model. The steady state of the model closely approximates that of the simulation. The dynamics during transitions are shown to be well modeled. For this process, the nonlinearity is due primarily to changes in gain rather than changes in dynamics; thus our nonlinear model is a good representation of the process.

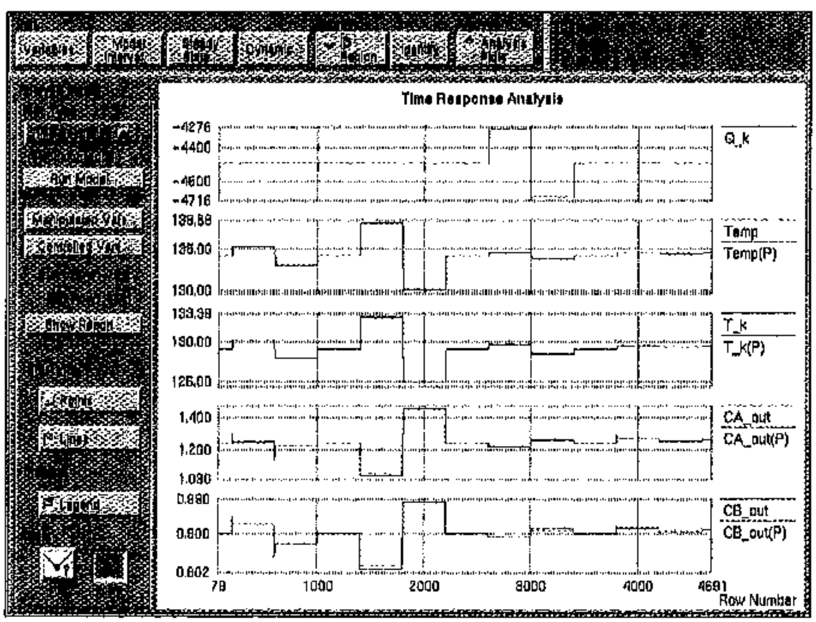

Figure 6. Comparison of the time response for the identified model versus the actual nonlinear plant model for identical excitation (i.e., MVs/DVs of which only Q_k is shown) to the model and the simulated system.

Because of the nonlinearity, controlling the process is challenging. According to [10], "the system is very difficult to control with a fixed time-invariant linear controller when only the concentration c_B is measured." In [10], a multiple controller approach was adopted because of the need "to compensate for the sign (gain) change of the plant." Using the multiple controller approach, according to [10], "the problem then is to determine when the controller sign must be changed." As shown next, our approach was able to overcome this problem by using a nonlinear controller.

Our nonlinear model was used in the MPC algorithm to control a simulation of the process. First, we examined the reference tracking capability of the controller. Referring to Figs. 7 and 8, at point A the desired output concentration, c_{Bout}, was set to 1.10 [mol/l] in a step change. The nonlinear controller achieved the new desired value in approximately 19 min. (The sampling interval of the controller was 3 min.) The multiple controller described in [10] showed a transition settling time of approximately 30 min. Both controllers were able to control the simulation through changes in gain.

With unavoidable variations in process inputs and operating conditions, robustness to exogenous disturbances is a critical feature for any control scheme. We therefore examined the robustness of the nonlinear controller to the exogenous disturbances (i.e., variations in disturbance variables). Referring to Fig. 7, the inlet flow was changed to 188.3 ±8 [m^3/h] (at points C and D) and held constant at these levels for 150 min. The nonlinear controller rejected the disturbances in inlet flow without significant variation in the desired concentration of the output product c_B. Note that only Q_k was used as the manipulated variable in this case. Rejecting variations in T_{in} and c_{Ain} requires both Q_k and F_{in} as manipulated variables. Therefore, from point F to point L in Figs. 7 and 8, the system was controlled with two manipulated variables. As shown in Figs. 7 and 8, while the change in inlet temperature, T_{in}, from 130 °C to 134 °C required only small changes in heat removal rate to preserve the output, c_B, at the set point, its change to 126 °C required a significant increase in inlet flow. After returning T_{in} to its predisturbed value of 130 °C, the inlet concentration, c_{Ain}, was changed to 5.1 ±0.3 [mol/l]. As Fig. 7 indicates, the disturbance in C_{Ain} was mainly canceled by control actions on heat removal rate Q_k. Thus the nonlinear controller was capable of providing fast control response, with considerable robustness to exogenous disturbances in a highly nonlinear problem.

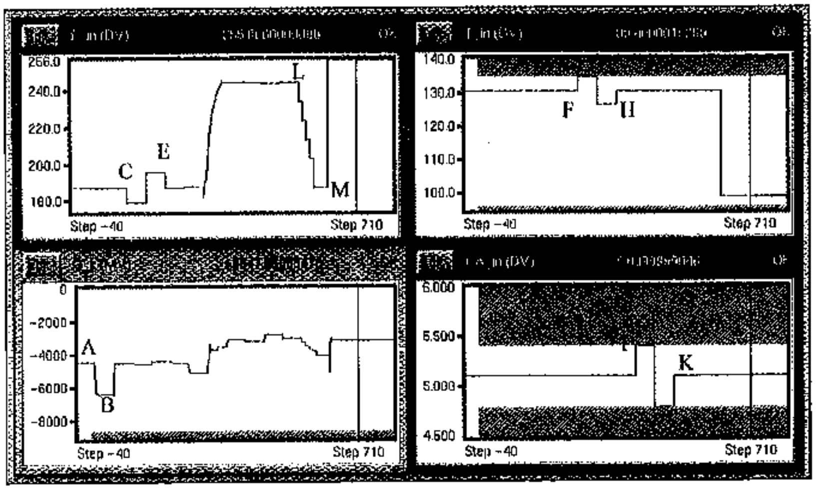

Figure 7. *Manipulated and disturbance variable profiles for the simulation scenario under the nonlinear controller. Note that from point F to point K the input flow rate is treated as MV.*

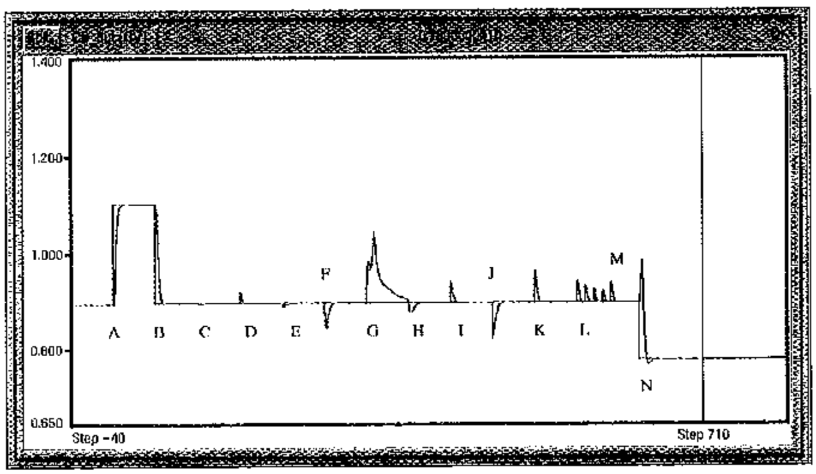

Figure 8. *The controlled variable, output concentration, under the nonlinear controller. Note that at A, a change in the output concentration is commanded. From point F to point L, the inlet flow is treated as a manipulated variable.*

Next, the ability to control major setpoint changes over nonlinear operating regions was examined. Referring to Figs. 7 and 8, starting at point L, we treated the inlet flow F_{in} as a disturbance variable again and subjected this variable to a sequence of step changes. Note that the controller maintained the output concentration at the desired value over these nonlinear transitions.

As a final test, transition of the output to a new setpoint, c_{Bout} = 0.78 [mol/l], was examined using heat removal rate as the manipulated variable. As Figs. 7 and 8 indicate, the nonlinear controller achieved the new desired output concentration in approximately 29 min.

As shown, the nonlinear model used in the MPC algorithm enabled the control of a challenging nonlinear process and showed that the nonlinear model was a good representation of the simulated process. Furthermore, we showed that the controller was capable of good reference and disturbance rejection control. Next, the results of using the controller on an actual industrial application are examined.

Polymer Application

The nonlinear model already described has been used in a wide variety of industrial applications, including Kamyr digesters (pulp and paper), milk evaporators and dryers (food processing), toluene diamine purification (chemicals), polyethylene and polypropylene reactors (polymers), gas- and coal-fired boilers (power), and a fluid catalytic cracking unit (refining). Highlights of a polyethylene application are given below [11].

The nonlinear controller has been applied to a gas phase polyethylene reactor at EC Erdolchemie GmbH in Cologne, Germany. The reaction produces low-, medium-, and high-density linear polyethylene over a wide range of melt indexes (MIs). The average production rate is 28 tons/h.

Optimal control of the process is difficult to achieve because the reactor is a highly coupled, nonlinear MIMO system. The main inputs to the process are the reactor hydrogen concentration ratio, the reactor comonomer ratio, the reactor temperature, and the catalyst efficiency. The primary outputs are MI, density, and production rate.

The nonlinear model used in the MPC algorithm enabled the control of a challenging nonlinear process and showed that the nonlinear model was a good representation of the simulated process.

Nonlinearity of a process can be measured in terms of change in gains over the operating region. For this process, the gains of several variable pairs vary over the operating region by a factor of 10 or more. The dynamic responses of the pairs do not change significantly over the operating region, but do vary between input-output pairs. Some variables have fast dynamics, while others have slow dynamics.

Significant process disturbances add to the difficulty of control. Catalyst activity, the impact of which is felt on all aspects of the reactor, is sensitive to many disturbances such as catalyst composition, activator concentration, temperature, and gas composition. Some of these variables affect catalyst activity at levels of parts per billion (ppb), making the system very sensitive to these disturbances. Other process disturbances include residence time, heat balance, temperature, and reliability. Finally, sensors for direct measurement of the main control variables, MI and density, are not available. Before beginning work on the controller, the main control variables were measured in a laboratory every 4 to 8 h.

The primary goal of the reactor controller is to produce the desired MI and density (strong importance), with maximum production rate (medium importance) and minimum consumption of raw materials (weak importance), while respecting heat removal, discharge, and agglomeration constraints (critical importance). The controller must allow fast transitions from one operating region to another while rejecting disturbances when operating at a single setpoint.

Transition control is important because the reactor is used to produce many grades of polymer. A transition from one product to another typically takes several hours to complete with a corresponding loss in prime production. Financial losses from each transition range from \$10,000-\$50,000. On average, the EC line is moved from one grade to another twice a week. Minimizing transition time has great economic benefit; however, it is difficult because of the inherent nonlinearities in the process. Prior to our work, the transitions were accomplished by a highly skilled operator because of the nonlinearities. Although a variety of different approaches for transition control have been tried on similar reactors such as linear and gain-scheduled controllers [12], [13], the results have been limited, and therefore transition control has relied on the skill of experienced operators.

The control system used for this application included our nonlinear MPC controller and an inferential model for online prediction of the primary controlled variables. The operator interacted with the control system by inputting the desired setpoints for melt index, density, production rate, and catalyst activity. The controller computed actuator setpoints for the reactor concentrations, catalyst and activator flows, as well as the residence time. Because sensors did not exist for the key controlled variables, neural-network-based inferential models were developed using historical lab data. The inferential models used the process inputs as well as occasional lab data to compute the online MI and density. The nonlinear steady-state model of the controller was constructed using historical data. The historical data contained examples of all products and production rates of interest, as well as known disturbance information. Accurate dynamic models were derived from both historical data and knowledge of the process.

The controller exhibited excellent disturbance rejection. The variance around the setpoint for each of the controlled variables was reduced by at least 50%. In addition, the average transition time has been significantly reduced. Fig. 9 shows two examples of a transition from an MI of 2.7 to 0.9. One transition is made by an operator, while the other is

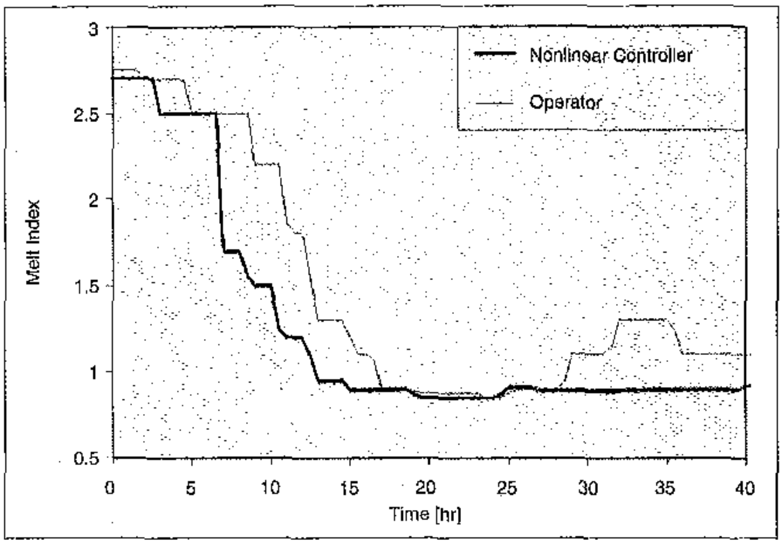

Figure 9. *Two MI transitions from 2.7 to 0.9. The first transition is made by the operator, while the second is made by the nonlinear controller. The nonlinear controller accomplishes the transition faster and holds the MI within specification.*

made by our nonlinear controller. Both transitions started at time 0. The controller completed the transition to specification in 12 h, while the operator completed the transition in 17 h. Thus the controller reduced transition time by approximately 30%. Notice that the controller kept the product on specification after the transition while the operator was not able to keep the product within specification. Finally, a reduction in raw material use and an increase in catalyst activity have been reported using the nonlinear controller [11].

Reduced transition time, variance reduction, and the ability to operate near process constraints have resulted in an increase in production rate ranging from 3-5%. The stability and control of the process have been improved to the point that record production rates have been achieved. Finally, there has been a significant decrease in abnormal control-related incidents, which in the past decreased production rates.

In addition to this application, the nonlinear control system has been successfully applied to 14 other polyolefin reactors at different manufacturing sites [14]. These applications use the nonlinear controller and estimators described above. All of the polyolefin reactor control systems show similar or better performance.

Conclusion

The MPC algorithm and associated nonlinear model described here present a unique solution to the control of coupled, constrained, nonlinear processes. By solving a constrained optimization problem over a future time horizon using a nonlinear model, challenging processes can be optimally controlled.

The nonlinear MPC controller presented in this article is created by combining a nonlinear steady-state model and a linear dynamic model. The nonlinear steady-state model is derived from historical data, while the linear dynamic model is created from process step test data. The two models are combined using a two-point gain scheduling technique that results in a parsimonious quadratic difference equation for each input-output pair. This model is a good representation of a process that has large changes in process gain.

The nonlinear controller provides excellent control in both simulation and industrial applications. In both cases, excellent disturbance rejection and transition control are observed. In an industrial polymer reactor application, the controller was shown to significantly reduce variance around desired output setpoints, reduce transition time, and increase overall production rate.

The nonlinear MPC controller presented has been recognized as the most widely used nonlinear MPC controller in the process industries [3]. It has achieved widespread use for three reasons:

- The primary nonlinearity in most processes, change in steady-state gain, is modeled. By restricting nonlinearity only to changes in gain, the nonlinear models can be systematically determined from historical and process step test data.
- The model is parsimonious, allowing for fast execution in an MPC-based algorithm.
- Software for model development, controller tuning, and run-time implementation is commercially available.

Acknowledgment

We would like to thank Edward Plumer and Celso Axelrud for their help in preparing this article.

References

[1] S.J. Qin and T.A. Badgwell, "An overview of industrial model predictive control technology," in *Chemical Process Control - AIChE Symposium Series*, J. Kantor, C. Garcia, and B. Carnahan, Eds. New York: AIChE, 1997, pp. 232-256.

[2] D.E. Seborg, "A perspective on advanced strategies for process control (Revisited)," in *Proc. European Control Conf.*, Karlsruhe, Germany, 1999.

[3] S.J. Qin and T.A. Badgwell, "An overview of nonlinear model predictive control applications," in *Proc. IFAC Workshop Nonlinear Model Predictive Control - Assessment and Future Directions*, Ascona, Switzerland, June 3-5, 1998.

[4] E.S. Meadow and J.B. Rawlings, "Model predictive control," in *Nonlinear Process Control*, M. Henson and D. Seborg, Eds. Englewood Cliffs, NJ: Prentice-Hall, 1997, pp. 233-310.

[5] S. Nash and A. Sofer, *Linear and Nonlinear Programming*. New York: McGraw-Hill, 1996.

[6] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing*, D. Rumelhart and J. McClelland, Eds. Cambridge, MA: MIT Press, 1986, pp. 318-362.

[7] E. Hartman, "Training feedforward neural networks with gain constraints," *Neural Computation*, to appear.

[8] L. Ljung, *System Identification*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[9] H. Chen, A. Kremling, and F. Allgöwer, "Nonlinear predictive control of a benchmark CSTR," in *Proc. 3rd ECC*, Rome, Italy, 1995, pp. 3247-3252.

[10] J. Trierweiler, "A systematic approach to control structure design," Ph.D. dissertation, Univ. Dortmund, 1996.

[11] M. Muller, "Optimization of a polyethylene gas-phase plant," in *Proc. Pavilion Tech. Users Conf.*, Austin, TX, 1999.

[12] J. Debling, G. Han, F. Kuipers, VerBurg, J. Zacca, and W.H. Ray, "Dynamic modeling of product grade transitions for olefin polymerization processes," *AIChE J.*, vol. 40, no. 3, pp. 506-520, 1994.

[13] J. MacGregor, A. Penlidis, and A. Hamielec, "Control of polymerization reactors: a review," *Polymer Process Eng.*, vol. 2, pp. 179-206, 1984.

[14] S. Goff, D. Johnson, and M. Gerules, "Nonlinear control and optimization of a high density polyethylene reactor," in *Proc. Chemical Engineering Expo*, Houston, TX, June, 1998.

Stephen Piché received his B.Sc. in electrical engineering from the University of Colorado, Boulder, in 1986. He received his M.Sc. and Ph.D. in electrical engineering from Stanford University in 1989 and 1992, respectively. From 1992 to 1995, he was a member of the technical staff at MCC in Austin, TX, where he designed and implemented VISA's neural-network-based credit card fraud detection system. From 1995 to early 2000, he was Director of Research at Pavilion Technologies in Austin, TX. While at Pavilion, he was also Chief Designer of Pavilion's model predictive controller. Currently, Dr. Piché is CEO and President of GlobeSolve.com, an Internet business-to-business e-commerce software and services provider located in Austin.

Bijan Sayyar-Rodsari received his B.Sc. in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1990. He received his M.Sc. and Ph.D. in electrical engineering from Stanford University in 1996 and 1999, respectively. Currently, he is a Research Scientist at Pavilion Technologies, Inc. His research interests include robust adaptive filtering and control, and modeling, optimization, and control of hybrid systems.

Doug Johnson received his B.Sc. in chemical engineering in 1981 from Texas A&M University. He has been with Pavilion Technologies, Inc., for four years and is currently the Technical Manager of the Polymers Group. He has 20 years of experience in polymer manufacturing and advanced process control.

Mark Gerules received his B.Sc. in mechanical engineering in 1987 and his M.Sc. in mechanical engineering in 1989 from the University of Kansas. He currently works as a Senior Consulting Engineer at Pavilion Technologies, Inc. His interests include the application of multivariable nonlinear optimizing control to the process industries.