Dynamically Scheduled MPC of Nonlinear Processes Using Hinging Hyperplane Models

Yugender Chikkula and Jay H. Lee
Dept. of Chemical Engineering, Auburn University, Auburn, AL 36849

B. A. Ogunnaike
E. I. Du Pont de Nemours and Co., Wilmington, DE 19880

This article deals with the issues associated with designing scheduled model predictive controllers for nonlinear systems within the multiple-linear-model-based control framework. The issues of model set generation from empirical data and closed-loop application of the generated model set are considered. A method of hinging hyperplanes is proposed as a way to construct a piecewise linear dynamic model, conducive to dynamic scheduling of linear MPC controllers. The design and implementation of dynamically scheduled MPC using a hinge function model are discussed, as well as its advantages. Alternate MPC formulations considered here require more computation, but utilize the hinge function model as a global model. Simulated examples of isothermal CSTRs and a batch fermenter are also presented to illustrate the proposed methodologies.

Introduction

In most of the chemical processes, nonlinearity is a rule rather than an exception. In addition to batch processes, continuous processes that are operated over wide ranges exhibit significant nonlinear behavior. As more and more processes adopt flexible operating strategies requiring transitions between multiple operating points, nonlinearity is becoming an integral part of many process control problems. Explicit consideration of underlying nonlinearity is becoming not only desirable, but necessary.

Over the past few decades, significant progress has been made in nonlinear control theory and synthesis procedures (Isidori, 1989; Kravaris and Kantor, 1990; Bequette, 1991; Rawlings et al., 1994). Two of the approaches that attracted the attention of most researchers in process control are: (1) geometric nonlinear control (feedback linearization) and (2) nonlinear model-predictive control. However, realization of the benefits from these developments has been lacking in practice so far. This lack of industrial enthusiasm can be attributed primarily to the lack of accurate nonlinear models needed to apply the above methods, and the complexity of resulting controllers.

On the other hand, gain scheduling which is conceptually simple and easy to implement has been the favorite methodology of control engineers to deal with process nonlinearity. Its main attraction can be attributed to its ability to incorporate linear design methods and linear analysis tools. There are various applications in process control problems where gain scheduling has been shown to result in improved control (Knoop and Perez, 1994; Gulaian and Lane, 1990; Shinskey, 1977). One drawback of the traditional gain scheduling methodologies is that controllers are typically *static* functions of the scheduling variables. The controller gains or other parameters are scheduled based on the *instantaneous* values of the scheduling variable and *past* values or the *dynamics* of the scheduled variable are not considered. The limitations of the *static* approach to gain scheduling have been recently brought forth in Jutan (1989), Reichert (1992), and Kwatra and Doyle III (1995). Jutan (1989) presented a scheduling strategy for PI controllers in which the gain is scheduled based

Correspondence concerning this article should be addressed to J. H. Lee who is currently at the School of Chemical Engineering, Purdue University, West Lafayette, IN 47907.

on not only the current feedback error, but the past values as well. Similarly, a dynamically scheduled controller was derived by Reichert (1992) for nonlinear time-varying systems. Kwatra and Doyle III (1995) present a dynamic scheduling framework based on the input-output linearization and the internal model control (IMC) formalisms. Though the importance and advantages of a dynamic gain scheduling framework are established in the above studies, all of them are either specific to the particular nonlinear process or limited by the control design methodology considered.

A control paradigm that combines the conceptual simplicity of design, ease of implementation characteristic to gain scheduling, and the performance characteristics of an advanced control technique can be successful in practice. Therefore, the scheduled control within the model-predictive control (MPC) formalism was considered here as a viable approach: that is, an MPC controller was designed based on a *set of models* scheduled as a function of process state variables that characterize the variations in process behavior. The focus is on *linear* model structures so that it can be implemented within the existing technology such as DMC. Thus, scheduled MPC based on *linear* models can be viewed as a methodology for smooth transition from the existing linear model-based control to nonlinear model-based control without invoking the complexity of solving a nonlinear optimization problem on-line. Scheduled MPC used to be carried out in different forms within the MPC framework. The local linearization-based methods, such as nonlinear QDMC (Garcia, 1984) in which a nonlinear model is linearized at every sampling time around the current state to recompute the dynamic matrix, can be conceptually classified as scheduled MPC. The multiple-model-based predictive control strategies proposed by Yu et al. (1992), Schott and Bequette (1994), Johansen (1994), and Arkun and Banarjee (1995, 1996) are a few of the other approaches to scheduled MPC.

Within the scheduled MPC framework, the model set generation is an important issue. A framework in which one can develop models from input-output data can be very attractive in this context. In this article, the method of *Hinging Hyperplanes* (Breiman, 1993) is proposed as an attractive technique for model set determination from empirical data. The Hinging Hyperplane (HH) method allows us to divide the system's dynamic variable space into several linear regimes, based on available data. What results is a piecewise linear dynamic model, which is perfectly suited for dynamic scheduling of linear controllers. Because of the local linear characteristics of the model, the controller implementations can be carried out within the existing MPC technology. This approach can be viewed as a hybrid between the often practiced gain scheduling and seldom-practiced, but widely investigated, nonlinear empirical model-based control. From an identification standpoint, the HH methodology practically combines some of the advantages of neural networks (Narendra and Parthasarathy, 1990; Bhat and McAvoy, 1990), such as the ability to handle large input dimensions and the fast computational properties of wave-nets (Bakshi and Stephanopoulos, 1994; Kouloris, 1995), offering an efficient way of constructing piecewise linear approximations for nonlinear systems. We evaluate the efficacy of the HH methodology in the context of nonlinear system identification and dynamic gain scheduled control within the MPC framework.

Given a hinge function model, MPC algorithms of varying computational complexities can be formulated. From a practical standpoint, the simplest formulation is the frozen-time linear time-invariant (LTI) formulation, which leads to dynamic scheduling of a set of linear MPCs. The computational complexity of this algorithm is essentially the same as that of the classical MPC. A slight extension to this is a linear time-varying (LTV) formulation in which a sequence of linear models (chosen on the basis of a previously computed input trajectory) are assumed in the prediction. The computational increase for this extension is quite mild. Finally, a rigorous nonlinear formulation that incorporates the global hinge function model directly into the prediction can be cast in the *propositional logic* framework (Tyler and Morari, 1996). This, however, requires solving a mixed integer quadratic programming (MIQP) problem, which significantly increases the computational burden. Issues related to each of these formulations are discussed.

A brief overview of gain scheduling and nonlinear empirical modeling is given. The HH methodology to nonlinear identification and implementation of *dynamically* scheduled model-predictive control based on a hinge function model are presented. Alternate MPC formulations for a hinge function model are also considered. The issues related to designing and implementing scheduled MPC using HH models are discussed. Applications to isothermal CSTRs and a batch fermenter problem are presented to illustrate the proposed framework.

Gain Scheduling

Gain scheduling control and empirical modeling of nonlinear systems for scheduled MPC is reviewed briefly.

Assume that the following model describes the underlying process dynamics

$$\dot{x}(t) = f[x(t), u(t)]$$
$$y(t) = h[x(t), u(t)]$$

The typical gain scheduling procedure for a nonlinear process involves:

- Selecting a set of operating points covering the range of operation of interest.
- For each of the selected operating points, construct a linear time-invariant approximation to the process and design a linear controller for each of the linear models.
- For intermediate operating conditions, *schedule/interpolate* the linear controllers.

Suppose the controller to be implemented has adjustable parameters denoted by θ. The adjustable parameters can be the controller tuning parameters (like the P/I/D parameters) or model parameters for a model-based controller. In the gain scheduling implementation, θ is described as a function of process variables. When $\theta(t)$ is given as a function of $y(t)$ and/or $u(t)$, we have *static* gain scheduling. When it is given as a function of $x(t)$, we have a *dynamic* gain scheduling. Note that this would require complete state knowledge (either through measurement or estimation). In most gain scheduling implementations, the relationship between process variables and controller parameters are specified as a piecewise

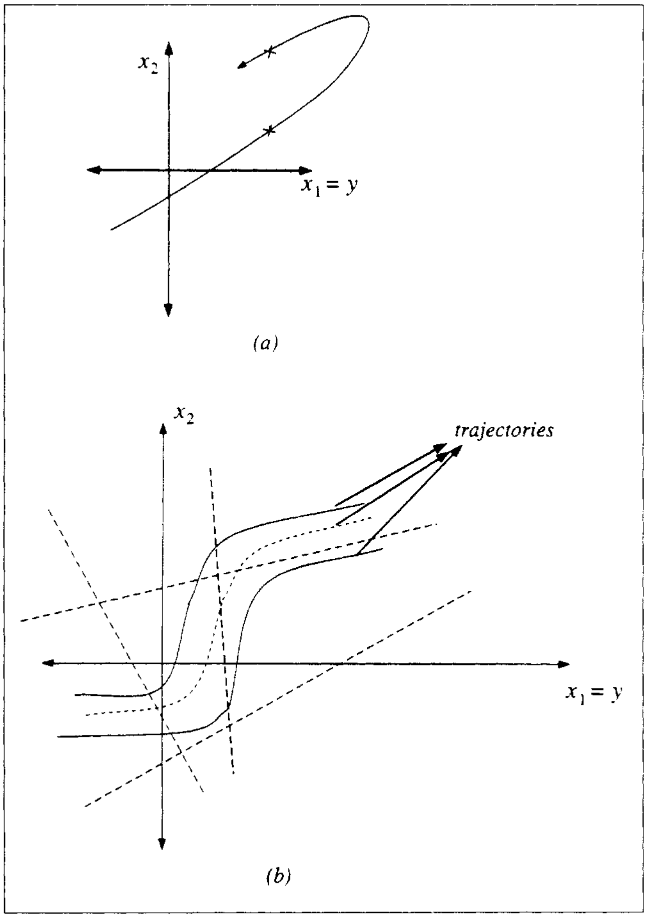

Figure 1. Static scheduling: limiting cases.

constant function; hence, the controller parameters need to be switched every once in a while whenever a change in the operating condition occurs.

Despite its practical success, the gain scheduling framework did not get the requisite attention from the academic community until recently (Shamma and Athans, 1990; Rugh, 1991). Very few analysis and synthesis results are available for the general gain scheduling problem and the scheduling is typically done based on intuition and experience, rather than through a systematic procedure. In most cases, static gain scheduling is used. In batch or other transition problems, linear models are developed at various points along the operating trajectory and used. In many cases, static gain scheduling can provide only limited performance. Some simple example cases are illustrated in Figure 1. As one can see, the current linear operating regime cannot always be determined by $y(t)$ alone. Dynamic gain scheduling, however, is seldom practiced, since it is very difficult to carry out on an *ad hoc* basis.

In predictive control context, some systematic methods are proposed in the literature. When a nonlinear model is available, Garcia (1984) proposed a nonlinear extension to QDMC based on successive linearization of the nonlinear model equations. Lee and Ricker (1994) later presented a state observer-based extension of the NLQDMC. Bequette and coworkers (1992, 1994) present a multiple model-based control scheme where the model switching in closed loop is carried out through a Bayesian criterion. Johansen and Foss (1994, 1995) developed an operating regime-based modeling and control framework for nonlinear processes. In this approach a set of local models and associated *validity* functions are identified using prior knowledge and input-output data from the process. Recently, Arkun and Banarjee (1995) and Banarjee and Arkun (1996) presented a multiple model-based identification and control strategy. In this approach they assume a set of models identified at various operating points and describe the process as a linear combination of these models in the state space. For intermediate operating conditions, the models are interpolated. These interpolation parameters are estimated on-line using a moving horizon type estimator. All these approaches, however, assume significant prior knowledge about the process which would facilitate regime decomposition or determination of appropriate locations for the local models.

Nonlinear Input/Output Modeling for Scheduled MPC

The difficulty of developing an accurate nonlinear model is thought to be the biggest obstacle to widespread use of nonlinear control methods (including dynamic gain scheduling). Since fundamental modeling is not feasible in most cases, models must be developed using plant input-output data. The input-output modeling of nonlinear systems has been widely investigated over the last few decades. The proposed approaches range from simple block-oriented approaches to more general approaches based on function approximation methods. Refer to recent surveys by Juditsky et al. (1996) and Pearson (1995) for an overview of the state of the art.

The problem of developing an empirical model involves inferring a mathematical relation from a given set of observations. For a single-input-single-output case the general problem can be stated as follows:

Let $D_N = \{u_1, \ldots u_N, y_1, \ldots y_N\}$ be a given set of input-output observations from a nonlinear system where $u_i \in R$ is the input data sequence and $y_i \in R$ is the output sequence. Find a functional relation $f{:}X \rightarrow y$ based on the data sequence D_N where X is the regressor vector to be determined from D_N.

The basic identification problem thus involves "concatenation of a mapping from observed data to a regression vector and nonlinear mapping from the regressor space to the output space" (Sjoberg et al., 1995).

The regressor or model-order determination can be approached in two ways: (1) using prior knowledge or (2) empirical determination from observed data. In a few cases the prior knowledge may be sufficient to specify the model order/regressor. However, in most cases the regressor vector may need to be determined from empirical data. There are many different ways to determine the particular regressor form such as methods based on information criteria and Rissanen's minimum distant length. The method of false nearest neighbors (Kennel et al., 1992; Abarbanel and Kennel, 1993; Rhodes and Morari, 1995) which determines the number of delayed inputs and outputs needed to construct a one-step-ahead predictor from a set of I/O data is also attractive in this context. With the knowledge of phase-space dimension, the identification problem can now be formulated as a function approximation problem in the determined phase space.

The input-output modeling can be regarded as synthesizing an approximation of a multidimensional function, a hypersurface reconstruction, within the chosen coordinate space. A wide variety of basis expansions have been investigated (Juditsky et al., 1996). Some of the widely reported approaches include: artificial neural networks (sigmoidal networks, radial basis function networks), wavelet networks, multivariate adaptive regression splines (MARS) (Friedman, 1991) and fuzzy models. The approximation properties of these methods are well studied (Barron, 1991; Juditsky et al., 1996). The choice between these approaches, however, is not clear and will largely depend on the particular application at hand.

Apart from the general basis function expansions, several other identification methodologies have been investigated. Since our focus in this work is scheduled MPC of nonlinear processes, we will consider some of the approaches that result in model structures that facilitate an implementation of scheduled MPC.

Regime Decomposition

Recently, Johansen and Foss (1995) have presented an algorithm for model set generation based on regime decomposition. In this approach they represent the system operating space as a d-dimensional box

$$Z = \left[z_1^{\min}, z_1^{\max}\right] \times \ldots \times \left[z_d^{\min}, z_d^{\max}\right]$$

where d is the dimension of the operating space, $z_1^{\min}$ and $z_1^{\max}$ are the ranges of the operating variable in the ith-direction. For a given set of observations spanning the operating regime, the algorithm *optimally* divides the operating space into different regimes and fits a *local* model structure from a given set of candidate model structures. The approach, though similar in spirit to MARS (Friedman, 1991) and other local modeling approaches, provides significant flexibility in implementation and facilitates incorporation of prior knowledge. However, one drawback of the approach is that the regime decomposition is considered parallel to the coordinate axis. More complex regime boundaries are not considered, and, as a result, may require a greater number of *local* models to represent a given system. Moreover, the algorithm is highly computer-intensive.

Parameterized Linearization Families

Consider a nonlinear system described by the following equations

$$\dot{x}(t) = f[x(t), u(t)]$$

$$y(t) = h[x(t), u(t)]$$

Linearizing the above system about a family of constant operating points (provided such a family exists), one can obtain the following family of linearized systems for the above nonlinear system

$$\dot{x}(t) = \frac{\partial f}{\partial x}(x(\tilde{u}),\tilde{u})[x - x(\tilde{u})] + \frac{\partial f}{\partial u}(x(\tilde{u}),\tilde{u})[u - \tilde{u}]$$

$$y(t) - y(\tilde{u}) = \frac{\partial h}{\partial x}(x(\tilde{u}),\tilde{u})[x - x(\tilde{u})] + \frac{\partial h}{\partial u}(x(\tilde{u}),\tilde{u})[u - \tilde{u}]$$

Wang and Rugh (1987) derived conditions for the existence of such families for a nonlinear system. The coefficient functions above are parameterized based on the input $\tilde{u}$. Alternate parameterizations based on output and state are also possible.

The above concept of a linearization family can be utilized in constructing systematic methodologies for nonlinear identification. In this approach the *local* time-invariant, linear models are identified at a set of *frozen* or *static* operating points along the constant operating trajectory. Then, these models are *combined* to form a nonlinear approximation to the underlying system. Perev et al. (1995) present a detailed analysis of the approximating properties of the resulting linearization family for the underlying nonlinear process. Though practically appealing, the approximation properties of the linearization family are strongly influenced by the *rate* of change of the operating point, the local curvature of the process and the *magnitude* of the input signal (Perev et al., 1995) since the linear models are identified under *small* signal conditions. In the absence of significant prior knowledge, determination of the number and location of the *local* linear models required for an acceptable approximation of the underlying nonlinear process is a nontrivial task.

Though the identification framework based on small signal conditions is attractive from a practical standpoint, the resulting model set is restrictive. Motivated by this, in this article we consider a nonlinear identification scheme with a global nonlinear structure. However, because of the intrinsic nature of the particular basis function used, the model structure facilitates local linear interpretations and, at the same time, is not limited by the small signal assumptions typical to most other local linear modeling frameworks.

Nonlinear Identification: Method of Hinging Hyperplanes

The function approximation approach to nonlinear identification can be cast as the following expansion over a set of basis functions

$$f(X) = \sum_{j=1}^{K} h_j(X) \qquad (1)$$

Various methods differ in the particular choice of the set of basis functions $\{h_j(X)\}$ used to represent the model space. In neural networks, we typically have the sigmoidal function and in wavelet networks the functions are drawn from a set of orthonormal wavelet basis. The identification problem then involves determination of the set of basis functions and the corresponding coefficients in the above expansion from system observations.

The Hinging Hyperplane (HH) method of Breiman (1993), considered in this work, also falls in the same category. In

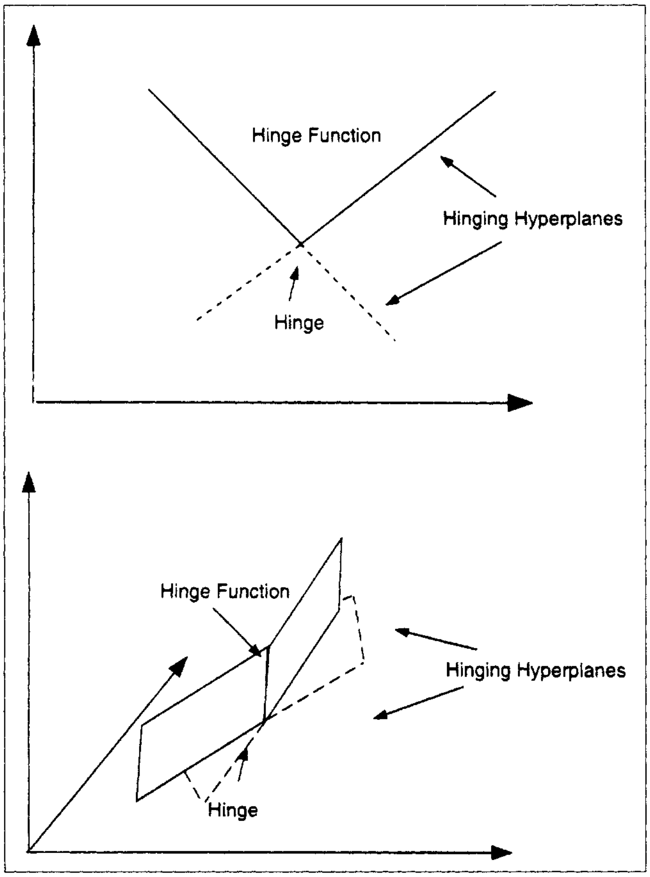

Figure 2. Hinge functions.

the HH method, the basis function we use is the following hinge function

$$h(X) = < \max|\min > (X^T\theta^+, X^T\theta^-) \qquad (2)$$

where $X=[1, x_1, x_2 \ldots, x_n]^T$ is the regressor vector. For example, in a time series description $x_1, \ldots, x_n$ are the past inputs $\{u_{k-i}\}$ and the past outputs $\{y_{k-i}\}$. The parameter vectors θ^+ and θ^- define the hyperplanes and are to be determined from the given data. At the *hinge location*, the two hyperplanes are joined and this location (subspace of the input space) is defined as: $\{X{:}X^T(\theta^+ - \theta^-)=0\}$. The joint, $\Delta = \theta^+ - \theta^-$, is defined as the *hinge* for the two hyperplanes. The hinge functions in one and two dimensions are shown in Figure 2.

Hinge function theorem

For a given system $f(X)$ satisfying the following smoothness assumption

$$\int \| \omega \|^2 \hat{f}(X) d\omega = c < \infty \qquad (3)$$

the upper bound on the approximation error with K single-knot hinge functions is given by

$$\| f(X) - \sum_{j=1}^{K} h_j(X) \| \leq \frac{2R^2 c}{K^{1/2}}$$

where R denotes the radius of the ball of approximation and $\hat{f}$ is the Fourier transformation of $f(X)$. This result essentially means that with sufficient number of hinge functions one can have an arbitrarily close approximation of any f in the smoothness class given in Eq. 3 within a given size ball. In addition it shows that the hinge models are as efficient as neural networks for the L_2-norm within the chosen smoothness class (Eq. 3) (Breiman, 1993; Juditsky et al., 1996).

Hinging hyperplanes: algorithms

Let us consider the problem of approximating the nonlinear function $f(X)$ by K *hinge* functions. We will give a brief description of the original hinge finding algorithm (HFA) attributable to Breiman (1993).

Breiman's HFA

Breiman's HFA involves iteratively fitting a hinge function $h_j(X)$ to a given set of data points. Assume that an identification experiment yielded an input-output data set $\{y_k, u_k\}_{k=1, \ldots, N_d}$ where N_d is the number of data points. The first step would be to determine the embedding dimension for the underlying dynamics using false nearest neighbors (FNN) method or any other equivalent method. Denote the resulting regressor vector by X.

When fitting K single-knot hinge functions for a given set of data, an algorithm called hinge finding algorithm (HFA) is repeatedly called which fits *one* single-knot hinge function at a time for a given set of data. The algorithm proceeds as follows:

- Assume an arbitrary hinge location $\Delta = \theta^+ - \theta^-$. Split the data set $\{y_k, X_k\}$ along the hinge direction. Denote the two data sets as

$$S_+ = \{X{:}X^T\Delta > 0\}$$

$$S_- = \{X{:}X^T\Delta \leq 0\}$$

- Estimate θ^+ by performing a least-square fit to the data set S_+ and the corresponding y_k. Similarly estimate θ^- by performing a least-square fit to the data set S_- and the corresponding y_k.
- Reconfigure the hinge location $\Delta = \theta^+ - \theta^-$ and update S_+ and S_- according to the new hinge location.
- Repeat the above process until the hinge function converges.

An important disadvantage of Breiman's algorithm is that it does not guarantee the global convergence of the hinge location. This aspect of Breiman's HFA can be better established by considering the following parameter estimation problem in HFA (Pucar and Sjoberg, 1995)

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} \sum_{k=1}^{N_d} [y_k - h(X_k, \theta)]^2 \qquad (4)$$

By dividing the parameter vector θ into two parts, each corresponding to one of the two hyperplanes, we have

$$\theta = \begin{bmatrix} \theta^+ \\ \theta^- \end{bmatrix}$$

Now the gradient and the Hessian for the above criterion (Eq. 4) can be shown to be as follows

$$\nabla V = \begin{bmatrix} \dfrac{\partial V}{\partial \theta^+} \\ \dfrac{\partial V}{\partial \theta^-} \end{bmatrix} = \begin{bmatrix} -\Sigma_{k \in S_+} X_k (y_k - h(X_k, \theta)) \\ -\Sigma_{k \in S_-} X_k (y_k - h(X_k, \theta)) \end{bmatrix}$$

$$\nabla^2 V = \begin{bmatrix} \Sigma_{k \in S_+} X_k X_k^T & 0 \\ 0 & \Sigma_{k \in S_-} X_k X_k^T \end{bmatrix}$$

The Newton's algorithm for solving Eq. 4 is

$$\begin{aligned} \theta_{\ell+1} &= \theta_\ell - (\nabla^2 V)^{-1} \nabla V \\ &= \theta_\ell + \begin{bmatrix} \left(\Sigma_{k \in S_+} X_k X_k^T\right)^{-1} \Sigma_{k \in S_+} X_k y_k - \theta_\ell^+ \\ \left(\Sigma_{k \in S_-} X_k X_k^T\right)^{-1} \Sigma_{k \in S_-} X_k y_k - \theta_\ell^- \end{bmatrix} \\ \theta_{\ell+1} &= \theta_\ell + \left(\theta_{\ell+1}^B - \theta_\ell\right) \end{aligned}$$

where θ_ℓ^B is the estimate from Breiman's HFA (linear least-square fit for each of the hyperplanes). The above result shows that Breiman's algorithm essentially is a Newton method and, as is well known, the Newton method is not *globally* convergent. Because of this, the initial hinge specification for the HFA is very critical. For a given initial hinge location, the algorithm can behave in three different ways (Pucar and Sjoberg, 1995): (1) converge to a hinge location within the data; (2) oscillate between a set of hinge locations; (3) diverge (that is, the hinge location is outside the given data range). However, utilizing this insight and reformulating the parameter estimation problem as a modified Newton method (Gradient-descent, Gauss-Newton, and so on) one can implement a *globally* convergent hinge finding algorithm thus making the HH methodology more practically appealing (Pucar and Sjoberg, 1995; Niu and Pucar, 1995).

Fitting multiple hinge functions

Initially, a single hinge function is fit and then the residuals are computed. In the next stage, another single-knot hinge function is fit for the residual space and the procedure is repeated until K hinge functions are fit. During the multiple hinge fit, the following *refitting* is carried out in order to reorient the hinge locations to the optimal directions.

- For $j = 1, \ldots, K$, find $h_j(X)$ to fit $f(X) - \Sigma_{i=1}^{j-1} h_i(X)$.
- Now refit

$$\begin{aligned} h_1(X) &\Rightarrow f(X) - \sum_{i=2}^{K} h_i(X) \\ h_2(X) &\Rightarrow f(X) - \sum_{i=1, i \neq 2}^{K} h_i(X) \\ &\vdots \\ h_K(X) &\Rightarrow f(X) - \sum_{i=1}^{K-1} h_i(X). \end{aligned}$$

- Repeat the above until convergence.

Discussion

Model structure

The HH methodology yields a model structure where for any given point in the X space we have a *linear* model which describes the local dynamics around that point. An important point to note is that this is different from the local linearization of a nonlinear model. That is, a *local linear* model from the HH approximation (which is fitted globally along with other local models) can be valid over a larger subset of the state space than local models fitted individually to local data around respective nominal operating points.

Connections to other modeling frameworks

The HH methodology can be interpreted in the artificial neural network framework as well. Instead of the sigmoid basis function, as is the case in feedforward networks, the HH methodology employs the hinge function as the basis function. The method is conceptually similar to multivariate adaptive regressor splines (MARS) as well and can be interpreted as a hybrid approach between basis function expansions (such as neural networks) and recursive partitioning approaches (such as MARS). In MARS, the splitting of the input space is carried out in the directions of the coordinate space. Hence, it is sensitive to the definition of the coordinates. In HH method, on the other hand, the input space is divided in arbitrary directions. This could potentially result in faster convergence and a smaller number of basis functions may be needed for a given approximation problem.

When considering a model structure for nonlinear system identification, a transparent interpretation of the resulting model can be useful in subsequent model utilizations. The neural network models involve linear projections of the input variables and subsequent nonlinear transformations which make them difficult to interpret. The wavelet networks, because of their local characteristics, are relatively more transparent. However, for higher dimensional problems the wavelet networks do not seem to be suitable. The recursive partitioning and tree-structured modeling approaches yield relatively more transparent models and in some cases can give insights into the underlying system structure. The HH method inherits this property of recursive partitioning methods and results in a transparent, piecewise linear approximation of a nonlinear system.

Set of local ARX models

Given process observations, the HH approach described yields a set of local ARX models provided the past inputs and outputs define the coordinate space. Compared to other local model approaches, the HH method differs in the model validity functions. In HH method the model validity function is the *indicator* function. It differs from the recent work of Johansen and Foss (1995), again in the way the input space is divided. They propose to split the input space along coordinate directions whereas HH method splits along arbitrary directions; this can potentially result in approximations with fewer basis functions.

Incorporation of prior knowledge

The HH methodology is a black-box approach to nonlinear identification. Despite this, prior information about mode changes can potentially be incorporated in the initial guess for hinge locations provided the chosen coordinate system facilitates such translation of the prior knowledge.

Model adaptation

Model adaptation can be formulated in a closed-loop setting. However, given the *nonorthogonal* nature of the basis function, we suggest that the model adaptation be carried out in an off-line fashion. Though all the model parameters need to be updated, the previous hinge locations will act as an excellent initial guess for the new identification step. Also, the linear least-square nature of the algorithm facilitates a computationally efficient implementation of the adaptation problem.

Model uncertainty

In HH framework, we have explicit knowledge of the number and quality of the data points that are used in fitting a particular *local linear* structure. This *qualitative* information can potentially be used in tuning the *local* controller corresponding to the model. For example, in MPC framework we can specify large/small input weighting for different local controllers and make the overall controller more/less aggressive in appropriate regions depending on the confidence level of the model.

MPC Using HH Models

Model interpretation

The attractive property of the HH methodology is that the resulting models are *piecewise* linear approximations to the underlying nonlinear system. Because of this, the HH models can be utilized in designing *linear* controllers for nonlinear systems within the general gain scheduling framework.

Consider a HH model structure where model output is expressed as a sum of K single-knot hinge functions as follows

$$y_k = \sum_{j=1}^{K} h_j(X_k)$$

where

$$h_j(X_k) = \langle \max | \min \rangle \quad \{X_k^T\theta_j^+, X_k^T\theta_j^-\}$$

Note that inclusion or exclusion of the max/min parts of the hyperplane are determined during identification stage and without loss of generality let us assume that *max* part of the hyperplanes are used to model the system. Then, for a given state X_k, the above predictor y_k can be written as

$$\begin{aligned} y_k = \quad & \{X_k^T\theta_1^+ \quad \text{if } X_k^T(\theta_1^+ - \theta_1^-) > 0 \\ & X_k^T\theta_1^- \quad \text{if } X_k^T(\theta_1^+ - \theta_1^-) \le 0\} \\ & + \{X_k^T\theta_2^+ \quad \text{if } X_k^T(\theta_2^+ - \theta_2^-) > 0 \\ & X_k^T\theta_2^- \quad \text{if } X_k^T(\theta_2^+ - \theta_2^-) \le 0\} \\ & \vdots \\ & + \{X_k^T\theta_K^+ \quad \text{if } X_k^T(\theta_K^+ - \theta_K^-) > 0 \\ & X_k^T\theta_K^- \quad \text{if } X_k^T(\theta_K^+ - \theta_K^-) \le 0\} \end{aligned} \tag{5}$$

Since all the regressors in each of the basis functions are the same for any given X_k, we can write the above model in the following form

$$y_k = X_k^T\tilde{\theta}_k$$

where

$$\tilde{\theta}_k = \sum_{i=1}^{K} \theta_i^{\pm}(X_k)$$

For an approximation with K hinge functions, we will have *at most* 2^K such possible linear models each of which describes the *local* dynamics of the system. The control problem can now be formulated as an MPC scheme with model switching based on Eq. 5. Note that because of the intrinsic structure of the model, the switching of either the controller/model parameters is based on the sequence of past inputs/outputs. Because of this, the resulting control scheme is *dynamically* scheduled.

Frozen-time LTI formulation: dynamic gain scheduling of linear MPC

Let us consider a multi-input-multi-output process where $y \in R^n$ is the output vector and $u \in R^m$ is the input vector. Assume that a HH fit of the identification data for the ith output gives a hinge function model consisting of K_i single-knot hinge functions as follows

$$y_k = \begin{bmatrix} y_{1,k} \\ y_{2,k} \\ \vdots \\ y_{n,k} \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{K_1} h_{1,j}(X_{1,k}) \\ \sum_{j=1}^{K_2} h_{2,j}(X_{2,k}) \\ \vdots \\ \sum_{j=1}^{K_n} h_{n,j}(X_{n,k}) \end{bmatrix} = \begin{bmatrix} X_{1,k}^T\tilde{\theta}_{1,k} \\ X_{2,k}^T\tilde{\theta}_{2,k} \\ \vdots \\ X_{n,k}^T\tilde{\theta}_{n,k} \end{bmatrix} \tag{6}$$

So, the dynamics of the ith output are described by at most $N_i = 2^{K_i}$ local linear models where the validity of each of the models is defined according to Eq. 5 with an appropriately defined state vector X_i. At time k, for a given set of state/regressor vectors $\{X_{i,k},\ i=1,\ \ldots,\ n\}$ denote the set of n local linear models valid for each of the outputs as $\vartheta_k = \{\tilde{\theta}_{1,k}, \tilde{\theta}_{2,k}, \ldots, \tilde{\theta}_{n,k}\}$.

A simple MPC formulation for the above hinge function model is the *linear-time-invariant* formulation with model switching, if needed, carried out at every sampling instant. For this, the controller problem can be formulated as the following minimization problem

$$\min_{\Delta u_k, \ldots, \Delta u_{k+M-1}} \sum_{\ell=1}^{P} \left(y_{k+\ell|k}(\vartheta_k) - r_{k+\ell}\right)^T \Lambda_y \times \left(y_{k+\ell|k}(\vartheta_k) - r_{k+\ell}\right) + \sum_{\ell=0}^{M-1} \Delta u_{k+\ell}^T \Lambda_u \Delta u_{k+\ell} \quad (7)$$

subject to

$$\begin{aligned} & y_{\max} \geq y_{k+\ell|k}(\vartheta_k) \geq y_{\min} \\ & u_{\max} \geq u_{k+\ell} \geq u_{\min} \\ & \Delta u_{\max} \geq |u_{k+\ell} - u_{k+\ell-1}| \quad \ell = 0, \ldots, k+M-1 \\ & u_{k+M-1} = u_{k+M} = \ldots = u_{k+P} \end{aligned}$$

where $y_{k+\ell}(\vartheta_k)$ is the predicted output vector at time $k+\ell$ based on measurements until time k and the model set ϑ_k consisting the *current* valid model for each of the outputs based on $\{X_{i,k},\ i=1,\ \ldots,\ n\}$. The prediction horizon P and control horizon $M \leq P$ along with the output weighting matrix Λ_y and the input weighting matrix Λ_u are the user specified tuning parameters. The objective is minimized with respect to M future input moves $\Delta u_k, \ldots, \Delta u_{k+M-1}$. As usual, only Δu_k is actually implemented on the plant.

Note that because of the intrinsic structure of Eq. 6 and the switching criterion (Eq. 5), the optimization in Eq. 7 is a quadratic programming problem. The control problem is solved in a receding horizon fashion, and only the first input move resulting from the solution of Eq. 7 is implemented. Once we have the new measurement, the model conditions (Eq. 5) are again evaluated for each output to determine the new set of prediction models ϑ_k. The model switching is based on the *sequence* of past inputs and/or outputs depending on whether we have a set of ARX or FIR models from the HH modeling, thus rendering the scheduling strategy its *dynamic* characteristic.

The control strategy is conceptually similar to the nonlinear QDMC approach suggested by Garcia (1984) in which a local linearization of nonlinear model equations around a present state is performed to determine local linear model parameters. A slight variation that would bring our approach even closer to the nonlinear QDMC approach is to use the HH model to predict the "free" response of the output (that is, the response when the input is held constant at the present value), which corresponds to the integration of nonlinear differential equations in the QDMC approach. The effect of deviations from the current value can then be evaluated using the current local linear model.

LTV formulation: successive linearization based MPC

A linear time-varying (LTV) MPC algorithm can also be formulated with the *model sequence* over the prediction horizon successively updated at every time-step based on the previously computed input trajectory (Lee and Ricker, 1994)

$$\min_{u_k, \ldots, u_{k+M-1}} \sum_{\ell=1}^{P} \left(y_{k+\ell|k}(\vartheta_{k+\ell}) - r_{k+\ell}\right)^T \Lambda_y \left(y_{k+\ell|k}(\vartheta_{k+\ell}) - r_{k+\ell}\right) + \sum_{\ell=0}^{M-1} \Delta u_{k+\ell}^T \Lambda_u \Delta u_{k+\ell} \quad (8)$$

subject to

$$\begin{aligned} & y_{\max} \geq y_{k+\ell|k}(\vartheta_{k+\ell}) \geq y_{\min} \\ & u_{\max} \geq u_{k+\ell} \geq u_{\min} \\ & \Delta u_{\max} \geq |u_{k+\ell} - u_{k+\ell-1}| \quad \ell = 0, \ldots, k+M-1 \\ & u_{k+M-1} = u_{k+M} = \ldots = u_{k+P} \end{aligned}$$

where $\vartheta_{k+\ell}$ is the linear model set used for the output prediction at time $k+\ell$ which is evaluated based on the state sequence resulting from a previously computed input sequence. This again results in a readily implementable quadratic programming problem. Note that iterations within a time step are also possible, but at an increased computational cost (Lee and Ricker, 1994).

HH model-based control: global nonlinear formulation

As discussed, the hinge model structure is a *global* nonlinear structure. A rigorous formulation utilizing the global nonlinear (piecewise linear) model can be formulated by incorporating the model dependence on X explicitly into prediction as constraints. The model switching scheme can be incorporated into the closed-loop objective in a *propositional logic* form (Tyler and Morari, 1996).

For notational simplicity, we will consider the single-input-single-output case in the following formulation. Extensions to a multi-input-multi-output case are straightforward. Noting that at *any given time*, the model parameter vector is evaluated according to the following switching criterion

$$\begin{aligned} \tilde{\theta} = \ & \left\{ \begin{array}{lll} \theta_1^+ & \text{if} & X_k^T(\theta_1^+ - \theta_1^-) > 0 \\ \theta_1^- & \text{if} & X_k^T(\theta_1^+ - \theta_1^-) \leq 0 \end{array} \right\} \\ & + \left\{ \begin{array}{lll} \theta_2^+ & \text{if} & X_k^T(\theta_2^+ - \theta_2^-) > 0 \\ \theta_2^- & \text{if} & X_k^T(\theta_2^+ - \theta_2^-) \leq 0 \end{array} \right\} \\ & \vdots \\ & + \left\{ \begin{array}{lll} \theta_K^+ & \text{if} & X_k^T(\theta_K^+ - \theta_K^-) > 0 \\ \theta_K^- & \text{if} & X_k^T(\theta_K^+ - \theta_K^-) \leq 0 \end{array} \right\} \end{aligned} \quad (9)$$

For a K hinge function model, denote the set of possible parameter vectors as $\tilde{\Theta} = \{\tilde{\theta}_i, i=1, \ldots, N\}$ where $N = 2^K$.

A general approach to incorporate piecewise linear models into an optimization is to assign a binary integer variable to each linear model and add additional constraints that would force the integer variable for the valid model (determined by other conditions) to be one and all others to be zero (Bemporad and Morari, 1998). Define the region $\mathfrak{X}_i$ as the region in the regressor space defined by X, in which a particular parameter vector θ_i is valid. Note that $\{\mathfrak{X}_i; i = 1, \ldots, N\}$ are disjoint sets. Now define $\{I_i; i = 1, \ldots, N\}$, a set of *binary variables*, as follows

$$I_i = \begin{cases} 1; & \text{if} \quad X \in \mathfrak{X}_i \\ 0; & \text{otherwise} \end{cases}$$

Since, at *any given time*, only one parameter vector is valid, the variables I_i must satisfy the following relation

$$I_1 + I_2 + \cdots + I_N = 1 \tag{10}$$

Also, note that each region $\mathfrak{X}_i$ can be described as a set of linear constraints

$$G^i X \leq H^i \tag{11}$$

Consider the following constraint

$$(G^i X - H^i) - C(1 - I_i) \leq 0 \quad \forall i, j \tag{12}$$

where C is a vector containing *suitably* large positive constants. By satisfying this constraint along with Eq. 10, note that one can guarantee that $I_i = 1$ if and only if $X \in \mathfrak{X}_i$ (Tyler and Morari, 1996).

However, with this approach, one would get $2^K \times P$ integer decision variables where P is the size of prediction horizon. With a large K, the optimization can quickly become intractable. In the particular case of HH models, we can take advantage of the way the model equations are written and reduce it down to $K \times P$. Consider the following constraints instead

$$X^T(\theta_j^+ - \theta_j^-) - C(1 - I_j) \leq 0 \tag{13}$$

$$-X^T(\theta_j^+ - \theta_j^-) - CI_j < 0, \quad j = 1, \ldots, K \tag{14}$$

It is easy to verify that, with the above constraints,

$$I_j = 0 \quad \text{if} \quad X^T(\theta_j^+ - \theta_j^-) > 0 \tag{15}$$

$$I_j = 1 \quad \text{if} \quad X^T(\theta_j^+ - \theta_j^-) \leq 0 \tag{16}$$

The model parameter vector can then be written as

$$\theta = \sum_{j=1}^{K} (1 - I_j)\theta_j^+ + I_j \theta_j^- \tag{17}$$

With the variables as defined above, we can now formulate the mixed integer programming problem with the model switching *inside* the horizon as follows

$$\min_{u_k, \ldots, u_{k+M-1}, \{I_{k,j}: j=1, \ldots, K\}, \ldots, \{I_{k+P,j}: j=1, \ldots, K\}} \sum_{\ell=1}^{P} \times (y_{k+\ell|k} - r_{k+\ell})^T \Lambda_y (y_{k+\ell|k} - r_{k+\ell}) + \sum_{\ell=0}^{M-1} u_{k+\ell}^T \Lambda_u u_{k+\ell} \tag{18}$$

subject to

$$\begin{aligned} y_{\max} &\geq y_{k+\ell|k} \geq y_{\min} \\ u_{\max} &\geq u_{k+\ell} \geq u_{\min} \\ \Delta u_{\max} &\geq |u_{k+\ell} - u_{k+\ell-1}| \\ &\ell = 0, \ldots, k + M - 1 \\ u_{k+M-1} &= u_{k+M} = \ldots = u_{k+P} \end{aligned}$$

$$\begin{aligned} X_{k+\ell}^T(\theta_j^+ - \theta_j^-) - C(1 - I_{k+\ell,j}) &\leq 0 \\ -X_{k+\ell}^T(\theta_j^+ - \theta_j^-) - CI_{k+\ell,j} &< 0, \\ j = 1, \ldots, K; \ \ell = 1, \ldots, P \end{aligned} \tag{19}$$

where the output prediction *inside* the horizon is evaluated as follows

$$y_{k-\ell|k} = X_{k+\ell}^T \left[\sum_{j=1}^{K} (1 - I_{k+\ell,j})\theta_j^+ + I_{k+\ell,j}\theta_j^- \right] + d_{k+\ell|k}$$

where $d_{k+\ell|k}$ is the disturbance estimate at time $k + \ell$ based on the measurements until k (depending on the nature of the model used, one may or may not need a separate disturbance term). These formulations clearly show the underlying problem that needs to be solved: It is a mixed integer programming problem which is nonconvex. With a slight reformulation, the above can be cast as a mixed integer quadratic program (MIQP). Some approaches to solving MIQPs have been recently discussed in Tyler and Morari (1996).

Issues in HH Model-Based Control

Some important theoretical as well as practical issues in designing scheduled MPC for nonlinear systems based on the hinge function models need to be addressed. Here we briefly describe a few of the important issues and discuss some of the approaches to address these issues.

Feasibility and stability in HH model-based control

When output constraints are included in the optimization problem, infeasibilities may occur. For on-line implementation, a systematic way to deal with such scenarios is needed. One simple solution would be to incorporate slack variables to soften the output constraints as is done in traditional MPC implementations. Softening the constraints ensures that the algorithm will keep running. However, one would like the algorithm to keep finding feasible solutions without softening if a feasible input trajectory indeed exists. In frozen-time LTI formulation of Eq. 7 and the LTV formulation of Eq. 8, this is not guaranteed because the model assumed for the prediction may not be valid for the entire prediction horizon. The nonlinear algorithm (Eq. 18), on the other hand, can avoid these infeasibilities, in principle. However, this requires a *sufficiently* large horizon and a global solution to the nonlinear optimization problem.

Globally, a hinge function model is a nonlinear model and all the *stability* results that are developed for the general nonlinear MPC problem apply to the nonlinear MPC formulation (Eq. 18). However, it also suffers from the same basic problems of all the other nonlinear MPC algorithms: the nonconvexity of the optimization problem because of which a global optimum cannot be guaranteed using a gradient-based optimization method. In fact, even the feasibility of the optimization can easily be lost because of this (Mayne, 1995). Moreover, infinite horizon cost cannot be evaluated easily in general. Because of these limitations, the standard approaches based on *infinite horizon* (Rawlings and Muske, 1993), *end constraint* (Keerthi and Gilbert, 1988; Mayne and Michalska, 1990) and *contraction constraint* (Yang and Polak, 1993; De Oliveira and Morari, 1997) are not practically suitable methods for guaranteeing closed-loop stability in the above HH model-based control.

For the general nonlinear MPC problem, the closed-loop stability and feasibility are still largely open theoretical issues (Morari and Lee, 1997). However, there are a few promising approaches such as hybrid/variable horizon MPC (Michalska and Mayne, 1993) and *quasi-infinite* horizon MPC (Chen and Allgöwer, 1996). These approaches are applicable to the above hinge function model-based strategy. In fact, the particular form of the hinge function model offers substantial advantages in terms of implementing the above strategies.

Design of a globally stabilizing hinge model-based controller can be approached from a robust control standpoint as well. The model form is similar to the *polytopic* model forms considered in robust MPC formulations (Kothare et al., 1995). With the hinge model, the calculation of a state feedback minimizing the worst case quadratic cost can be formulated as linear matrix inequalities (LMIs) and a convex programming problem, similar to the one presented for polytopic model forms in Kothare et al. (1995). This approach, though readily applicable in the present context, can potentially result in conservative solutions, especially for highly *nonlinear* systems. This is because unlike the *uncertain* LTV case, the model parameters in a hinge model-based strategy do not vary arbitrarily with time (that is, they are functions of states).

Effects of disturbances

HH models, just as any empirical model, can be sensitive to disturbances, particularly those that were not reflected in the identification data. This issue is more closely tied to the choice of intrinsic model structure rather than that of basis functions. In this sense, HH models should not be any more or less sensitive to disturbances than models fitted with other basis functions. At issue is how well the disturbance effects are described by the stochastic part of the model so that model predictions remain accurate even when the disturbances do occur. What makes this a particularly difficult and relevant issue for nonlinear systems is that, for nonlinear systems, the effect of inputs and disturbances cannot be separated in general. Henceforth, any nonlinear modeling method that does not consider the unmeasured disturbance effects in a rigorous explicit manner can lead to problems. The NARX structure we considered in this article does include some unmeasured disturbance effects that are coupled with the input effects (through the autoregressive terms). A NARMAX structure that includes past residuals in the regressor vector would be even more general, but leads to some additional difficulties in model fitting. The issue of unmeasured disturbances, although very important, is general to nonlinear empirical modeling and is beyond the scope of this article.

Effects of noise

In Breiman's approach the hinge location (the input space where $\Delta X = 0$) is considered to be part of either one of the hyperplanes. Note that this is essential for the continuity of the hinge function model. However, in practice, because of noise effects, the switching scheme may fluctuate between models when the process is close to one of the hinge locations. In order to avoid this one can include a *dead zone* type modification to the model structure as follows

$$
\begin{aligned}
\tilde{\theta}_{k+1} = \quad & \left\{ \begin{array}{ll} \theta_1^+ & \text{if} \quad X^T(\theta_1^+ - \theta_1^-) > \epsilon \\ \theta_1^- & \text{if} \quad X^T(\theta_1^+ - \theta_1^-) \le -\epsilon \\ \theta_{1,k} & \text{if} \quad -\epsilon < X^T(\theta_1^+ - \theta_1^-) < \epsilon \end{array} \right\} \\
+ & \left\{ \begin{array}{ll} \theta_2^+ & \text{if} \quad X^T(\theta_2^+ - \theta_2^-) > \epsilon \\ \theta_2^- & \text{if} \quad X^T(\theta_2^+ - \theta_2^-) \le -\epsilon \\ \theta_{2,k} & \text{if} \quad -\epsilon < X^T(\theta_2^+ - \theta_2^-) < \epsilon \end{array} \right\} \\
& \quad \vdots \\
+ & \left\{ \begin{array}{ll} \theta_K^+ & \text{if} \quad X^T(\theta_K^+ - \theta_K^-) > \epsilon \\ \theta_K^- & \text{if} \quad X^T(\theta_K^+ - \theta_K^-) \le -\epsilon \\ \theta_{K,k} & \text{if} \quad -\epsilon < X^T(\theta_K^+ - \theta_K^-) < \epsilon \end{array} \right\} \qquad (20)
\end{aligned}
$$

Here $\theta_{i,k}$ is the parameter vector corresponding to the active segment of the hinge function i at time k.

Numerical Examples

Now let us consider numerical examples to demonstrate the approximation properties of the Hinge Function models and their application in scheduled MPC of nonlinear processes. We will consider two isothermal CSTR processes and a batch fermenter for this purpose.

Example 1: isothermal CSTR

Consider the isothermal CSTR process described by the following dynamic equation, originally from Bruns and Bailey (1975) and later adapted by Gattu and Zafiriou (1992) and Koulouris (1995)

$$\frac{dy}{dt} = u - y - \beta \frac{1}{1 + \frac{k_1}{y} + \frac{y}{k_2}} \tag{21}$$

Here y is the reactant concentration, and u is the feed reactant concentration. k_1 and k_2 are the kinetic parameters and β is a constant. The single enzyme-catalyzed reactions with substrate inhibited kinetics and ethylene hydrogenation in an isothermal CSTR can be described by the above model. With $k_1 = 0.01$, $k_2 = 0.1$, and $\beta = 2$, the process exhibits multiple steady states for a range of input values, of which two are stable and one is unstable.

Note that Eq. 21 is a first-order ODE and for a given initial condition at time $k-1$ on the output (y_{k-1}) and with a constant input u_{k-1} over a time interval $[k-1, k]$, the solution for y_k can be obtained by integrating Eq. 21. In other words, the above system can be embedded in a two-dimensional delay coordinate space (phase-space) consisting $\{y_{k-1}, u_{k-1}\}$. The system surface in the phase-space coordinates is shown in Figure 3. As can be seen, the system consists of a flat region corresponding to the low intensity equilibrium points and a sharp transition to the regions of high activity and exhibits significant nonlinear behavior.

Model identification

An input-output experiment was performed and the distribution of the I/O data in phase-space is shown in Figure 4. In order to elicit both low as well as high frequency behavior of the process, a probability based switching scheme was utilized to design the input signal. The input sequence generation is performed as follows

- $\{\psi_i, i = 1 \ldots\}$, a random sequence, we generated where ψ_i are i.i.d. with a uniform distribution between $[0,2]$.
- $u(1) = \psi(1)$.
- Choose $u(i) = \psi(i)$ with probability $1 - p_s$ and $u(i) = u(i-1)$ with probability p_s.

p_s is the switching probability and determines the nature of the generated signal. For large values ($p_s \approx 1$), the signals are of low frequency in nature and, conversely, for small values (p_s close to zero) we obtain signals of high frequency. A total of 1,000 data points were collected using the above probability switching scheme. The first 400 points were generated with $p_s = 0.9$, the next 400 points with $p_s = 0.5$ and the next 200 points with $p_s = 0.1$.

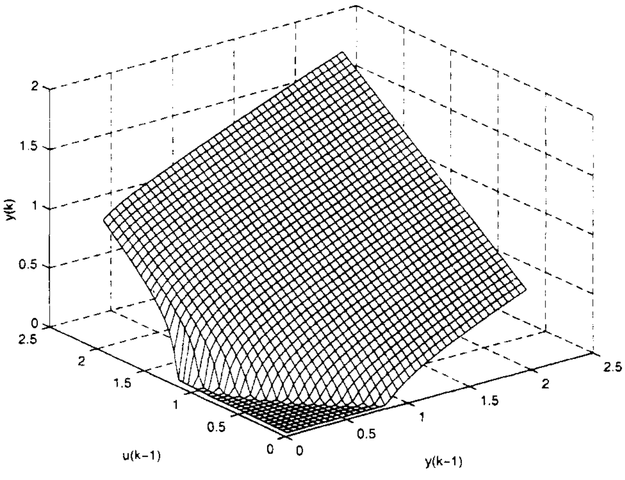

Figure 3. System surface in delay coordinates y_{k-1} and u_{k-1}: CSTR 1.

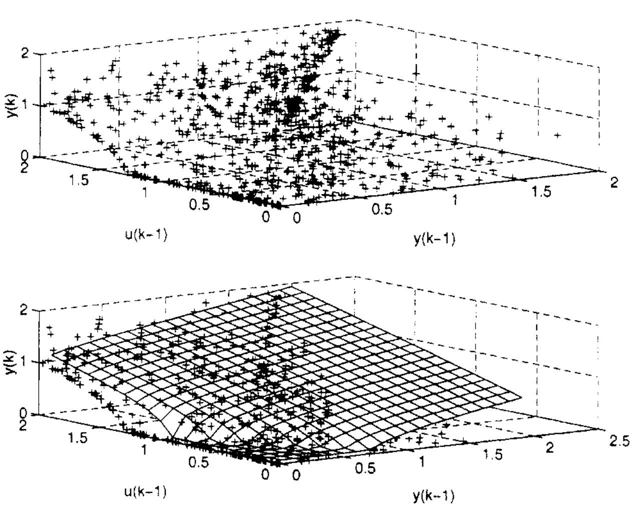

Figure 4. Distribution of I/O data for CSTR 1.

In the next stage, a hinge function model consisting of four single-knot hinge functions was used to approximate the system. The model parameters estimated from the I/O data are as follows

$$\begin{aligned} y_k = {} & \max\{-0.0001 + 0.0066u_{k-1} - 0.0050y_{k-1}, -0.4406 \\ & + 0.6947u_{k-1} + 0.5899y_{k-1}\} + \max\{0.1249 - 0.1293u_{k-1} \\ & - 0.1657y_{k-1}, 0.0150 - 0.0246u_{k-1} + 0.3193y_{k-1}\} \\ & + \min\{0.1099 - 0.0048u_{k-1} - 0.4850y_{k-1}, -0.1064 \\ & + 0.0038u_{k-1} + 0.3045y_{k-1}\} + \min\{0, -0.0052 \\ & + 0.0024u_{k-1} + 0.1998y_{k-1}\} \end{aligned}$$

Figure 5 demonstrates the single-knot hinge function approximations of the system and the corresponding error surface.

A global linear model was also identified using the same data set and the linear model parameters are obtained to be as follows

$$y_k = -0.2670 + 0.5508u_{k-1} + 0.5557y_{k-1} \tag{22}$$

Figure 6 gives a comparison of the predictive performances of the linear model and the HH model. As can be seen, the HH model gives better predictions compared to the linear model especially close to the origin in the input space. This should be expected since the HH model is sensitive to the transition from high to low activity and, accordingly, the local linear model valid to that region is used in the prediction. Note that close to the unstable steady state (region of large

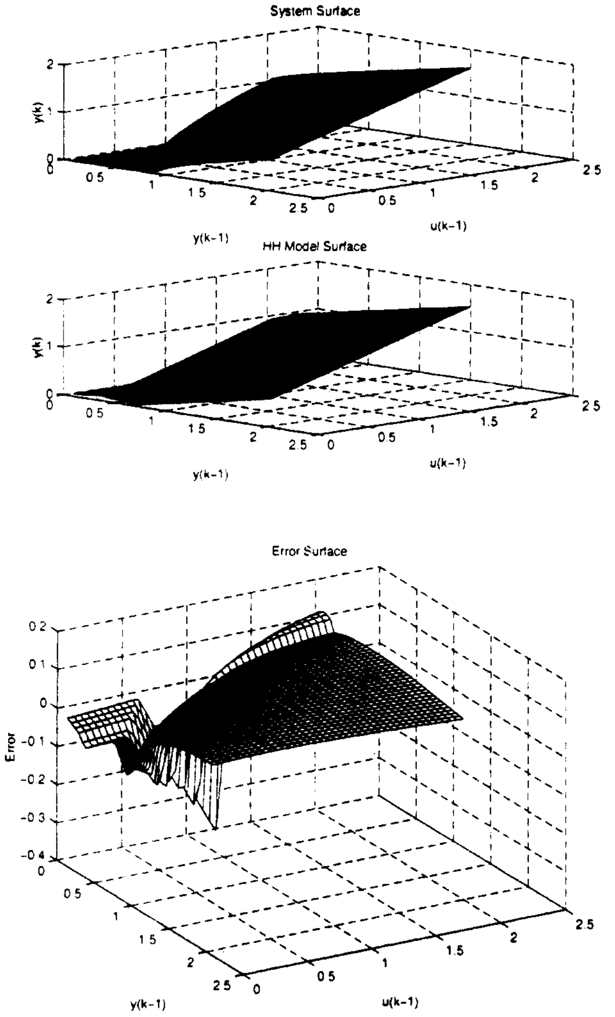

Figure 5. HH modeling of CSTR 1.

curvature), the model error is relatively high. This can be attributed to the lack of dynamic information around the unstable steady state, in the data obtained from the identification experiment.

Scheduled MPC of isothermal CSTR 1

Now let us consider a transition problem from the unstable region of the CSTR to the stable low activity region in example 1. We will consider an HH model with one hinge function, the first basis function in the model we obtained above.

$$y_k = \max\{-0.0001 + 0.0066u_{k-1} - 0.0050y_{k-1}, -0.4406 + 0.6947u_{k-1} + 0.5899y_{k-1}\}$$

Two MPC controllers are considered each with identical tuning parameters (shown in Table 1) differing only in the model used in the closed-loop predictions; the global linear model and the hinge function model.

The performance of the two controllers is shown in Figure 7 for a step change in the set point from 0.3 to 0.001. As can be seen, the closed-loop performance of the scheduled MPC is superior to the MPC based on the linear model. This is due to the inability of the global linear model to give good predictions in the low activity region. The hinge function model, on the other hand, is sensitive to the transition from the high to low activity regions and, accordingly, uses the local linear model corresponding to that region. The integral sum of the derivations from the set point for both controllers are given in Table 2.

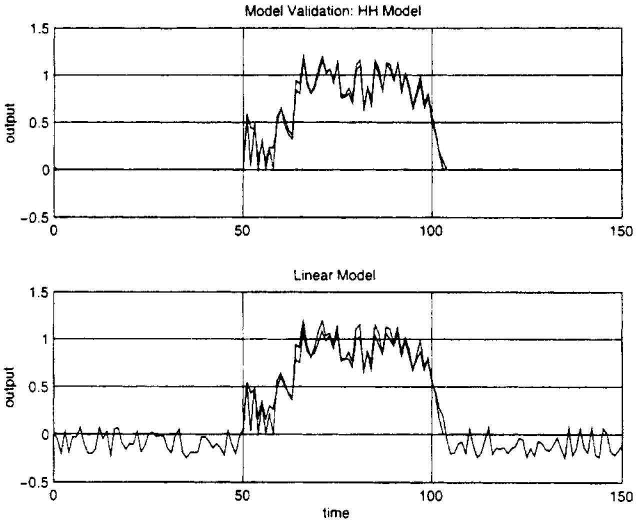

Figure 6. Model predictions: HH vs. linear model for CSTR 1.

Example 2: isothermal CSTR 2

Let us consider another isothermal CSTR process of different dynamics (Kwatra and Doyle III, 1995). The first-order reaction $A \rightarrow B$ is carried out in the CSTR and the reactor dynamics are described as

$$\frac{dC_A}{dt} = -kC_A + (C_{A_i} - C_A)\frac{q}{V} \tag{23}$$

In state-space form the above system can be represented as

$$\dot{x} = -kx + (C_{in} - x)u \tag{24}$$

$$y = x \tag{25}$$

Table 1. Controller Settings for CSTR 1

Controller	Parameters
p	4
m	2
λ_y	100
λ_u	0.1
$y_{\max}$	2
$y_{\min}$	0
$u_{\max}$	2
$u_{\min}$	0
$\Delta u_{\max}$	0.5

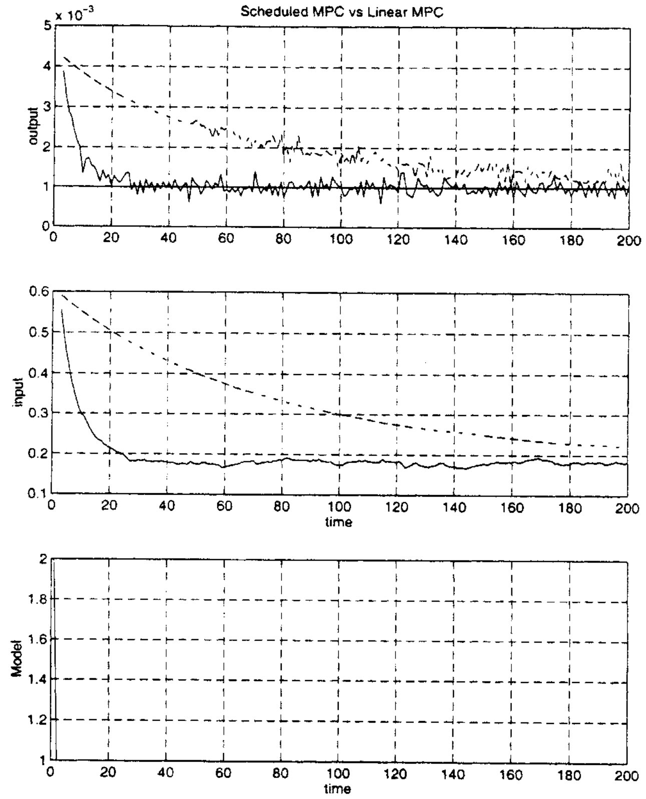

Figure 7. Linear MPC (---) vs. HH model-based control (——).

Here C_A is the concentration of A, C_{A_i} the inlet concentration of A, k the kinetic rate constant, q the inlet flow rate, and V the reactor volume. The process parameters were chosen to be $k = -0.028$ and $C_{in} = 1$.

Model identification

An input-output experiment, with input excitations generated similar to example 1, was performed and the resulting input-output data is shown in Figure 8. A hinge function model consisting of three single-knot hinge functions with the following structure was constructed

$$y_{k+1} = \max\{\theta_1^+ X_k, \theta_1^- X_k\} + \max\{\theta_2^+ X_k, \theta_2^- X_k\} + \min\{\theta_3^+ X_k, \theta_3^- X_k\}$$

$$X_k = [1 \ y_k \ u_k]^T$$

The model parameters are given in Table 3. The system and model surfaces along with the error surface are shown in Figure 9. Again, as is illustrated in the phase-space, the constructed Hinge model results in excellent approximation of

Table 2. Controller Performance: CSTR 1

MPC Controller	Integral Square Error
With global linear model	3.5098×10^{-4}
With hinge function model	4.6408×10^{-5}

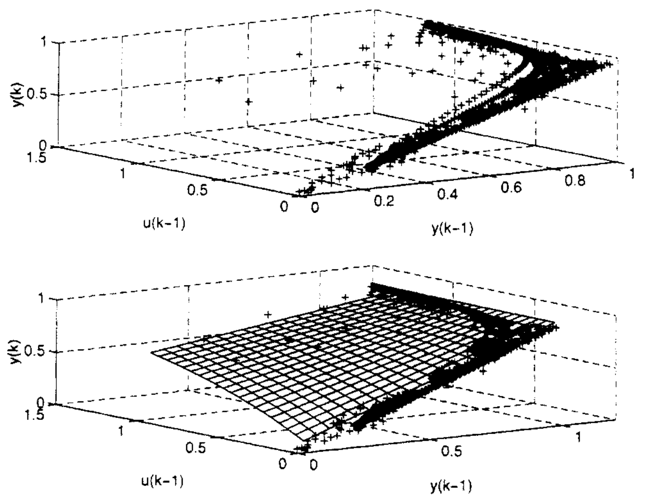

Figure 8. Input-output data for CSTR 2.

Table 3. Model Parameters: CSTR 2

Parameters	Values
θ_1^+	[0.0195 0.3024 0.9347]
θ_1^-	[0.4523 0.2282 0.4808]
θ_2^+	[0.1110 0.0040 −0.1205]
θ_2^-	[0.5096 0.0808 0.4087]
θ_3^+	[−0.4903 0.3806 0.5142]
θ_3^-	[0.0024 0.0782 −0.0104]

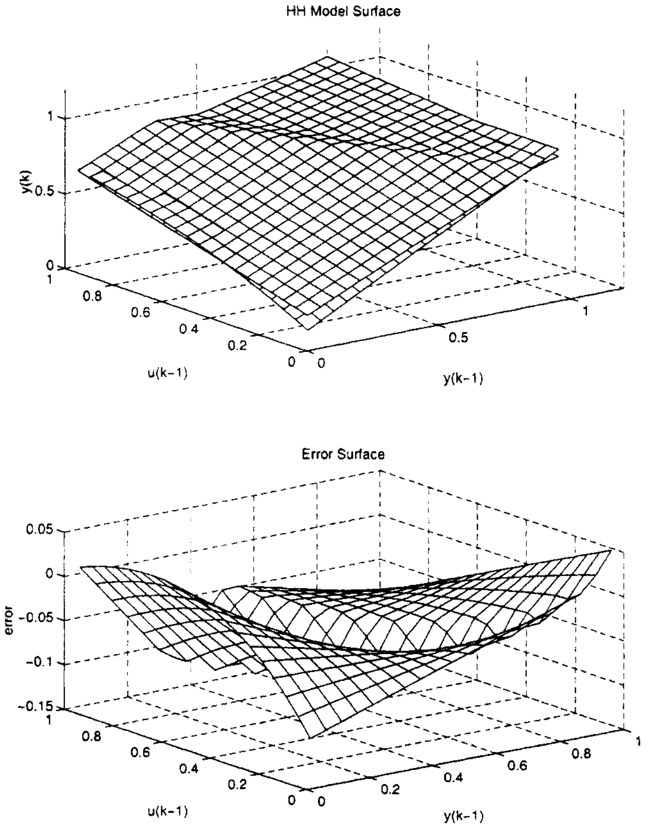

Figure 9. HH approximation to CSTR 2.

Table 4. Operating Conditions for Local Linear Models: CSTR 2

Model No.	Steady-State Input	Steady-State Output
1	0.0051	0.1541
2	0.0101	0.2651
3	0.0201	0.4179
4	0.0300	0.5172
5	1.9900	0.9861

the underlying nonlinear system surface. However, by visual inspection we note that the hinge location does not seem to be *optimally* located. This can be attributed to the fact that the data set contains very few observations corresponding to *large* input values and *small* output values. Despite this, however, note that the resulting model captures the *qualitative* behavior in this region.

Dynamically scheduled MPC of isothermal CSTR 2

Now the HH model is implemented in scheduled MPC framework and we will compare the performance of the *dynamically* scheduled MPC based on Hinge models with *static* scheduling of MPC on a set of *static* local linear models. The set of *static linear* models is obtained by linearizing the system equations at 5 different operating points. The steady-state operating points around which the linearization is performed are given in Table 4. Five model predictive controllers are considered based on these five local linear models with identical controller parameters except the input weighting which is tuned for improving the robustness of the local controller. The controller parameters are given in Table 6. The static scheduling is based on the *instantaneous* value of the output and the switching criterion is given in Table 5.

Figure 10 compares the performance of the *dynamically* scheduled and *static* scheduled model-predictive controllers for various set point variations. As is evident, the *dynamically* scheduled MPC based on the HH model outperformed the *static* scheduled controller over the entire operating range. Specifically for the set point variation from 0.95 to 0.65, the region where the system exhibits significant nonlinearity (curvature), the static scheduling does not give acceptable performance. For the output value around 0.95, the *valid* linear model according to the static scheduling scheme employed (model 5) has very small gain. However, the process has a relatively large gain around the output value of 0.65. When the MPC controller computes the input change required for the set point variation from 0.95 to 0.65 with the prediction model as model 5, because of the lower gain in the prediction model used, the input change computed is very large and input is driven almost to zero (Figure 10). However, as the system shifts close to an output value of 0.65, according to the static scheduling, the prediction model is switched to model 4. Since model 4 has a large steady-state gain, in the next control computation the model predicts the overcompensation and increases the input from the previous value computed according to model 5. This again drives the system back into the range of model 5 and the closed-loop system starts oscillating.

Table 5. Output Bounds for Static Scheduling of Local Linear Models: CSTR 2

Model No.	Output
1	$0.00 < y_k \leq 0.20$
2	$0.20 < y_k \leq 0.40$
3	$0.40 < y_k \leq 0.50$
4	$0.50 < y_k \leq 0.75$
5	$0.75 < y_k \leq 1.00$

Table 6. Controller Settings for Static and Dynamic Scheduled Controllers: CSTR 2

Parameters	Static Scheduling	Dynamic Scheduling
p	6	6
m	4	4
λ_y	1	1
λ_u	[0.2, 0.2, 3, 3, 3]	[0.2, 0.2, 0.2, 0.2, 0.2]
$y_{\max}$	2	2
$y_{\min}$	0	0
$u_{\max}$	2	2
$u_{\min}$	0	0
$\Delta u_{\max}$	0.5	0.5

Though the above closed-loop response can be made more acceptable by employing techniques such as *dead-zone* for switching, and so on, the inherent limitation of static scheduling based on a set of *static* linear models is evident from the above example. If the process has significant variations in local curvature, in absence of prior knowledge which would help in appropriate locations for linear model estimation and scheduling, *static* scheduling can potentially result in poor performance. On the other hand, the HH model, a global structure, captures these variations in process response surface and facilitates dynamically scheduled MPC.

Effects of disturbances and noises

We have also performed a number of simulations with random noises added at the output (uniformly distributed between 0–0.01 and 0–0.05) and in the inlet concentration

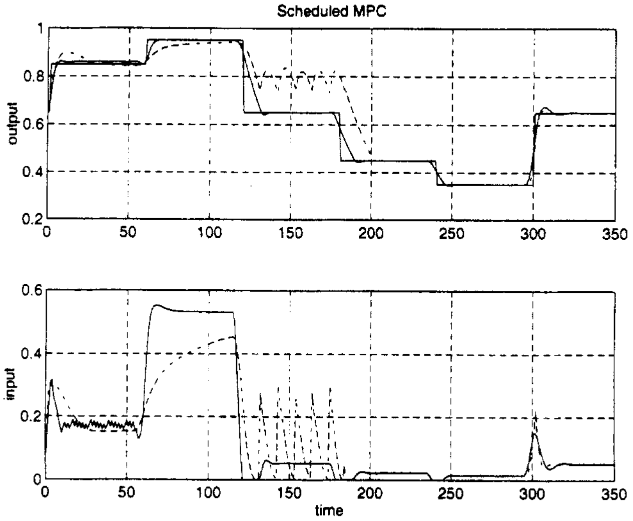

Figure 10. Dynamic (——) vs. static (---) scheduled MPC of isothermal CSTR 2.

(0–0.05). The hinge locations were observed to be different for each of the models and, depending on the noise level, the predictive capability of the model was also observed to vary. This should be expected of any data-driven modeling approach. However, the qualitative nature of the models was consistent and did not change with the noise level of the identification data. The closed-loop simulations showed satisfactory performances for all the models.

Example 3: batch fermenter

This is a biochemical system and describes the fermentation of glucose to gluconic acid in a batch STR. The process is described by the following dynamic equations

$$\dot{x}_1 = \mu_m \frac{x_1 x_4 x_5}{K_s x_5 + K_0 x_4 + x_4 x_5}$$

$$\dot{x}_2 = v_L \frac{x_1 x_4}{K_L + x_4} - 0.9082 K_p x_2$$

$$\dot{x}_3 = K_p x_2$$

$$\dot{x}_4 = -\frac{1}{Y_s} \mu_m \frac{x_1 x_4 x_5}{K_s x_5 + K_0 x_4 + x_4 x_5} - 1.011 \frac{x_1 x_4 x_5}{K_s x_5 + K_0 x_4 + x_4 x_5}$$

$$\dot{x}_5 = k_l a (x_5^* - x_5) - 0.09 v_L \frac{x_1 x_4}{K_L + x_4} - \frac{1}{Y_0} \mu_m \frac{x_1 x_4 x_5}{K_s x_5 + K_0 x_4 + x_4 x_5}$$

Here x_1 is the cell concentration, x_2 is glucanolactone concentration, x_3 is the gluconic acid concentration, x_4 is glucose concentration, and x_5 is dissolved oxygen concentration. The parameters μ, K_L, v_L and K_p are observed to depend on the temperature and pH. The set points to temperature and pH control loops are the manipulated variables $u = [\text{Temp pH}] = [u_1 \ u_2]$. For details of the model parameters and operating conditions, refer to Ghose and Ghosh (1976) and Johansen (1994).

For identification purposes, we assume that all the states are *measured.* The following one-step-ahead prediction forms are considered

$$x_1(k+1) = f_1\{x_1(k), x_2(k), x_3(k), x_4(k), x_5(k), u_1(k), u_2(k)\}$$

$$x_2(k+1) = f_2\{x_1(k), x_2(k), x_3(k), x_4(k), x_5(k), u_1(k), u_2(k)\}$$

$$x_3(k+1) = f_3\{x_1(k), x_2(k), x_3(k), x_4(k), x_5(k), u_1(k), u_2(k)\}$$

$$x_4(k+1) = f_4\{x_1(k), x_2(k), x_3(k), x_4(k), x_5(k), u_1(k), u_2(k)\}$$

$$x_5(k+1) = f_5\{x_1(k), x_2(k), x_3(k), x_4(k), x_5(k), u_1(k), u_2(k)\}$$

A total of fifty batch runs were simulated with different startup values of glucose and cell concentrations. The temperature and the pH were varied in a random fashion across the batch run.

Next, we estimated the Hinging Hyperplane models for each of the state evolutions utilizing the identification data. The obtained Hinge models are of the following form

$$x_1(k+1) = \max\{\theta_{11}^+ X_k, \theta_{11}^- X_k\} + \min\{\theta_{12}^+ X_k, \theta_{12}^- X_k\}$$

$$x_2(k+1) = \max\{\theta_{21}^+ X_k, \theta_{21}^- X_k\} + \max\{\theta_{22}^+ X_k, \theta_{22}^- X_k\}$$

$$x_3(k+1) = \max\{\theta_{31}^+ X_k, \theta_{31}^- X_k\} + \max\{\theta_{32}^+ X_k, \theta_{32}^- X_k\}$$

$$x_4(k+1) = \max\{\theta_{41}^+ X_k, \theta_{41}^- X_k\} + \min\{\theta_{42}^+ X_k, \theta_{42}^- X_k\}$$

$$x_5(k+1) = \max\{\theta_{51}^+ X_k, \theta_{51}^- X_k\} + \max\{\theta_{52}^+ X_k, \theta_{52}^- X_k\}$$

where

$$X_k = [1 \ x_1(k) \ x_2(k) \ x_3(k) \ x_4(k) \ x_5(k) \ u_1(k) \ u_2(k)]$$

On-line optimization of batch fermenter using HH models

Here we consider the application of the HH model in scheduled MPC of the fermenter for maximizing the gluconic acid production over a fixed batch time of $T_f = 5$ h. The following optimization is solved after every 0.5 h interval. At time k, we solve

$$\max_{U_k, \ldots, U_{T_f}} X_3(T_f)$$

subject to the constraints on the inputs: temperature and pH

$$\begin{bmatrix} 25 \\ 5.4 \end{bmatrix} \le \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \le \begin{bmatrix} 35.4 \\ 7 \end{bmatrix}$$

For comparative purposes, a linear model is also considered. The resulting batch profiles are compared in Figure 11. The average gluconic acid production rate over the batch pe-

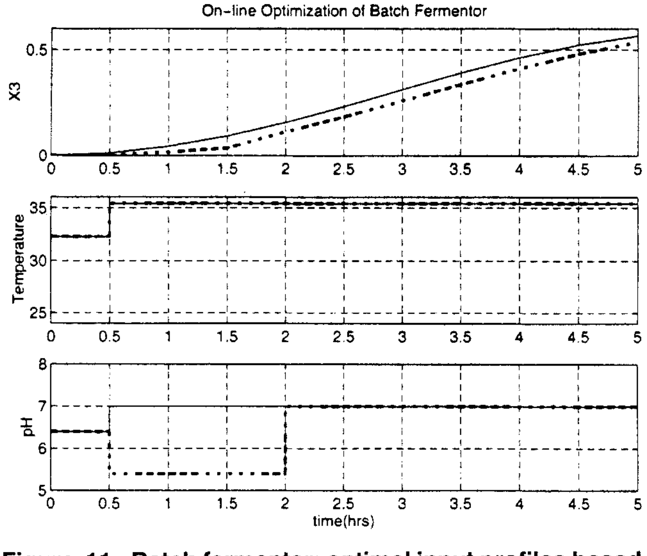

Figure 11. Batch fermenter: optimal input profiles based on linear (---) and HH models (——).

Table 7. Model Performance for Batch Fermenter

Model	Avg. Prod. Rate of Gluconic Acid
HH	5.65 g/lh
Linear	5.36 g/lh

riod is given in Table 7. As can be seen, the HH model captures the underlying nonlinearity and the improved performance is evident in the enhanced efficiency of the batch operation compared to the linear model-based on-line optimization.

Conclusions

In this article we considered *dynamically* scheduled model-predictive control as a practically implementable control strategy for nonlinear processes. Within the scheduled MPC framework, the problem of empirical identification of *local* linear models is addressed. We established the relevance and attractiveness of a new modeling paradigm *Hinging Hyperplane* methodology for this purpose. The hinge model structure is a global structure with local linear interpretations and is shown to facilitate practically implementable control strategies within the MPC framework. We also presented a predictive control formulation that utilizes the hinge model as a global nonlinear model. Simulation examples of chemical processes are presented in order to demonstrate the proposed methodology.

Acknowledgment

The authors gratefully acknowledge the support of DuPont, AspenTech and the National Science Foundation (grant CTS-9357827).

Literature Cited

Abarbanel, H. D. I., and M. B. Kennel, "Local False Nearest Neighbors and Dynamical Dimensions from Observed Data," *Phys. Rev. E*, **47**, 3057 (1993).

Arkun, Y., and A. Banarjee, "Robust Multiple Model Based Control of Nonlinear Systems," AIChE Meeting, Miami Beach, FL (1995).

Bakshi, R. B., and G. Stephanopoulos, "Wave-Net: A Multiresolution, Hierarchical Neural Network with Localized Learning," *AIChE J.*, **39**, 57 (1993).

Banarjee, A., and Y. Arkun, "Model Predictive Control of LPV Systems," AIChE Meeting, Chicago (1996).

Barron, A. R., "Universal Approximation Bounds for Superpositions of a Sigmoidal Function," Tech. Rep. No. 58, Dept. of Statistics, Univ. of Illinois at Urbana-Champaign (1991).

Bemporad, A., and M. Morari, "Predictive Control of Constrained Hybrid Systems," *Int. Symp. on Nonlinear MPC: Assessment and Future Directions*, Ascona, Switzerland (1998).

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

Bhat, N., and T. J. McAvoy, "Use of Neural Networks for Dynamic Modeling and Control of Chemical Process Systems," *Computers and Chem. Eng.*, **14**, 573 (1990).

Breiman, L., "Hinging Hyperplanes for Regression, Classification and Function Approximation," *IEEE Trans. Info. Theory*, **39**(3), 999 (1993).

Bruns, D. D., and J. E. Bailey, "Process Operation Near an Unstable Steady-State Using Nonlinear Feedback Control," *Chem. Eng. Sci.*, **30**, 755 (1975).

Chen, H., and F. Allgöwer, "A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability," *Automatica*, in press (1996).

De Oliveira, S. L., and M. Morari, "Contractive Model Predictive Control for Constrained Nonlinear Systems," *IEEE Trans. on Autom. Control*, in press (1997).

Friedman, J. H., "Multi-Variate Adaptive Regression Splines (with Discussion)," *The Annals of Statistics*, **19**, 1 (1991).

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

Gattu, G., and Z. Zafiriou, "Nonlinear Quadratic Dynamic Matrix Control with State Estimation," *Ind. Eng. Chem. Res.*, **31**(4), 1096 (1992).

Ghose, T. K., and P. Ghosh, "Kinetic Analysis of Gluconic Acid Productions by *pseudomonas ovalis*," *J. Appl. Chem. Biotechnol.*, **26**, 768 (1976).

Gulaian, M., and J. Lane, "Titration Curve Estimation for Adaptive pH Control," *Proc. of Amer. Cont. Conf.*, San Diego, 1414 (1990).

Hernandez, E., "Control of Nonlinear Systems Using Input-Output Information," PhD Thesis, Ga. Inst. of Technol., Atlanta (1992).

Isidori, A., *Nonlinear Control System—An Introduction*, Springer-Verlag, New York (1989).

Johansen, T. A., "Operating Regime Based Process Modeling and Identification," Dr. Ing Thesis, NTH, Univ. of Trondheim, Norway (1994).

Johansen, T. A., and B. A. Foss, "Identification of Nonlinear System Structure and Parameters using Regime Decomposition," *Automatica*, **31**, 321 (1995).

Juditsky, A., H. Hjalmarsson, A. Benveniste, B. Delyon, L. Liung, J. Sjoberg, and Q. Zhang, "Nonlinear Black-Box Models in System Identification," *Automatica*, in press (1996).

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

Keerthi, S., and E. Gilbert, "Optimal Infinite Horizon Feedback Laws for a General Class of Constrained Discrete-time Systems: Stability and Moving-Horizon Approximations," *J. of Optimiz. Theory and Appl.*, 265 (1988).

Kennel, M. B., R. Brown, and. H. D. I. Abarbanel, "Determining Embedding Dimension for Phase-Space Reconstruction Using a Geometrical Construction," *Phys. Rev. A*, **45**, 3403 (1992).

Knoop, M., and J. Perez, "Nonlinear PI-controller Design for a Continuous-Flow Furnace via Continuous Gain Scheduling," *J. Process Cont.*, **4**, 143 (1994).

Kothare, M. V., V. Balakrishnan, and M. Morari, "Robust Constrained Model Predictive Control Using Linear Matrix Inequalities," IFA-Report No. 95-02, Cal. Inst. of Technol., Pasadena (1995).

Koulouris, A., "Multiresolution Learning in Nonlinear Dynamic Process Modeling and Control," PhD Thesis, Dept. of Chemical Engineering, Mass. Inst. of Technol., Cambridge (1995).

Kravaris, C., and J. C. Kantor, "Geometric Methods for Nonlinear Process Control," *Ind. Eng. Chem. Res.*, **29**, 2295 (1990).

Kwatra, H. S., and F. J. Doyle, III, "Dynamic Gain Scheduled Control," AIChE Meeting, Miami Beach, FL (1995).

Lee, J. H., and N. L. Ricker, "Extended Kalman Filter Based Nonlinear Model Predictive Control," *Ind. Eng. Chem. Res.*, **33**, 1530 (1994).

Mayne, D. Q., "Optimization in Model Based Control," *Preprints IFAC Symp. DYCORD*, Hesingor, Denmark (1995).

Mayne, D. Q., and H. Michalska, "Receding Horizon Control of Nonlinear Systems," *IEEE Trans. on Autom. Cont.*, **35**, 814 (1990).

Michalska, H., and D. Q. Mayne, "Robust Receding Horizon Control of Constrained Nonlinear Systems," *IEEE Trans. on Autom. Cont.*, **38**(11), 1623 (1993).

Morari, M., and J. H. Lee, "Model Predictive Control: Past, Present and Future," *Preprint* (1997).

Narendra, K. S., and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. Neural Networks*, **NN 1**, 4 (1990).

Niu, S. S., and P. Pucar, "Hinging Hyperplanes for Nonlinear Identification," *Technical Report*, Linkoping Univ., Sweden (1995).

Pearson, R. K., "Nonlinear Input/Output Modeling," *Preprints, ADCHEM*, Kyoto, Japan (1994).

Perev, K., C. Jacobsen, and B. Shafai, "An Algorithm for Identification of Smooth Nonlinear Systems Based on Parameterized Linearization Families," *Int. J. Cont.*, **61**, 1013 (1995).

Pucar, P., and J. Sjoberg, "On the Hinge Finding Algorithm for Hinging Hyperplanes," *Technical Report*, Linkoping Univ., Sweden (1995).

Rawlings, J. B., and K. R. Muske, "The Stability of Constrained Receding Horizon Control," *IEEE Trans. Autom. Cont.*, bf 38, no. 10, 1512 (1993).

Rawlings, J. B., E. S. Meadows, and K. R. Muske, "Nonlinear Model Predictive Control: A Tutorial and Survey," *Preprints, ADCHEM*, Kyoto, Japan (1994).

Reichert, R., "Dynamic Scheduling of Modern Robust Control Autopilot Designs for Missiles," *IEEE Cont. Sys.*, 35 (1992).

Rhodes, C., and M. Morari, "Determining the Model Order of Nonlinear Input/Output Systems Directly from Data," *Proc. of ACC*, Seattle, WA (1995).

Rugh, W., "Analytical Framework for Gain Scheduling," *IEEE Cont. Sys. Mag.*, **11**, 79 (1991).

Schott, K. D., and W. B. Bequette, "Control of Nonlinear Chemical Processes Using Multiple Model Adaptive Control," AIChE Meeting, San Francisco (1994).

Shamma, J. S., and M. Athans, "Analysis of Gain Scheduled Control of Nonlinear Plants," *IEEE Trans. on Automatic Control*, **35**, 898 (1990).

Shinskey, F. G., *Distillation Control: For Productivity and Energy Conservation*, McGraw-Hill, New York, p. 258 (1977).

Sjoberg, J., Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear Black-Box Modeling in System Identification: A Unified Overview," *Preprint* (1995).

Tyler, M. L., and M. Morari, "Propositional Logic in Control and Monitoring Problems," Technical Report AUT96-15, ETH, Zurich, Switzerland (1996).

Wang, J., and W. J. Rugh, "Parameterized Linear Systems and Linearization Families for Nonlinear Systems," *IEEE Trans. on Circuits and Sys.*, **34**, 650 (1987).

Yang, T. H., and E. Polak, "Moving Horizon Control of Nonlinear Systems with Input Saturation, Disturbances and Plant Uncertainty," *Int. J. Control*, **58**(4), 875 (1993).

Yu, C., R. J. Roy, H. Kaufman, and B. W. Bequette, "Multiple-Model Adaptive Predictive Control of Mean Arterial Pressure and Cardiac Output," *IEEE Trans. Biomed. Eng.*, **39**, 448 (1992).

Manuscript received July 7, 1997, and revision received Sept. 8, 1998.