Paper 256h

Model-on-Demand Model Predictive Control for Nonlinear Process Systems

Martin W. Braun[†], Daniel E. Rivera[†], and Anders Stenman[††]
[†]Department of Chemical and Materials Engineering
Control Systems Engineering Laboratory,
Arizona State University, Tempe, Arizona 85287-6006

[††]Division of Automatic Control, Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden

(480) 965-9476
email: daniel.rivera@asu.edu, martin.braun@asu.edu, anders.stenman@nira.se

Prepared for presentation at the Annual AIChE Meeting, Los Angeles, CA
November 12-17, 2000
Session: Advances in Process Control II, Paper 256h

November, 2000

UNPUBLISHED

Model-on-Demand Model Predictive Control for Nonlinear Process Systems

M. W. Braun[†], D. E. Rivera[†,*], and A. Stenman[††]

[†]Department of Chemical and Materials Engineering
Control Systems Engineering Laboratory,
Arizona State University, Tempe, Arizona 85287-6006

[††]Division of Automatic Control, Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden

November 10, 2000

Abstract

An identification and control methodology is developed for nonlinear process systems that integrates the collection of prior knowledge, input signal design, model validation, and model predictive control. The methodology makes specific use of Model-on-Demand (MoD) estimation to predict nonlinear process dynamics. Previous work has shown MoD to provide comparable performance to other nonlinear modeling techniques, while requiring less engineering effort to train and validate the model. As a result of the integrated methodology, the cost, time, and effort required to develop a nonlinear model predictive control strategy is greatly reduced. Model-on-Demand Model Predictive Control (MoDMPC) can incorporate traditional linear MPC elements for stabilizing control action, and output constraint infeasibility handling. The advantages of the methodology are demonstrated on a non-adiabatic CSTR simulation, a nonlinear rapid thermal processing (RTP) reactor, and a non-minimum phase, block-structured Hammerstein process.

[*]To whom all correspondence should be addressed. phone: (480) 965-9476 fax: (480) 965-0037; e-mail: daniel.rivera@asu.edu

1 Introduction

Nonlinear control provides the opportunity for high performance control action over a wider operating range in comparison to linear control. However, the user would prefer to use nonlinear control methods that require engineering effort comparable to that required for linear control techniques, since model development has been estimated to account for 75% of the costs of designing an advanced control project (Hussain, 1999). This paper presents an approach that addresses the practical need for systematic, user-friendly, nonlinear control methodologies which avoid the computational difficulties of the full nonlinear predictive control problem and the arbitrary modeling decisions required in current local model control methodologies.

Model Predictive Control using local models has been explored in the process control literature in the context of fuzzy modeling, artificial neural networks, and other interpolation techniques. Fuzzy modeling ideas require the user to decide on how to partition the regressor space for good performance of local linear models. For most applications, this is a highly iterative process although a few automated methods have recently emerged. Neural network based approaches can be tedious since the user must judiciously select good initial parameter estimates. If the initial parameters are successful such that the neural network converges, the user must still decide whether the neural network structure is adequate and repeat the initial parameter/structure selection problem all over again.

"Model-on-Demand" (MoD) estimation provides an attractive alternative for nonlinear identification, since MoD requires less structural decisions to be made by the user. MoD is a hybrid between local and global modeling. MoD adaptively selects the number of data in the regressor space to construct a local model. The MoD algorithm continually constructs new local models as the current operating point moves about the regressor space. As a result, the initialization, optimization and structure selection issues of global semiphysical and neural network modeling are avoided. The user is not faced with decisions of how to partition the regressor space, commonly found with other local modeling techniques. Moreover, MoD can be implemented in an MPC framework to provide natural local model transition throughout the regressor space (MoDMPC). Other MPC controllers which use local linear models, must decide how to transition between the local linear models (e.g. Bayesian weighting, or interpolation). With MoD, the user need only determine the NARX regressor structure, and a lower bound on the number of data to be used in the local models. The best fit global linear ARX structure often provides a good choice for the NARX structure for MoD, and the lower bound on the number of data to be used for the local model can be determined through validation in 2 or 3 iterations.

This paper is organized as follows. Section 2 outlines the methodology and discusses the advan-

tages of Model-on-Demand estimation. Section 3 discusses the specific MoD formulation used in the methodology and the corresponding user decisions. Section 4 reviews the MoDMPC formulation proposed by Stenman (1999*a*; 1999*b*). In this section, modifications are made to the original formulation to provide improved performance such as stable control of unstable plants, intuitive handling of output constraint infeasibilities, and feedforward capabilities as mentioned above. Performance enhancements brought by these modifications are demonstrated on a CSTR simulation in the subsequent section. Effective output constraint infeasibility handling is demonstrated on a Hammerstein model exhibiting inverse response dynamics. Tuning is also shown to be a consideration in effectively handling the output constraint infeasibilities. Lastly, MoDMPC is shown to handle MIMO control of an RTP semiconductor wafer reactor, by keeping the wafer temperature profile within $\pm 0.5\ K$ across the wafer during a $50\ K/s$ temperature ramp rate. The paper concludes with a summary of the ideas presented and a discussion of opportunities for further research.

2 Model-on-Demand Identification and Control Methodology Overview

This paper proposes a Model-on-Demand Identification and Control methodology. The goal of this methodology is to provide the user with a structured approach to nonlinear identification and control, that maximizes use of *a priori* information, while minimizing the number of iterations between the steps within the methodology. The methodology is illustrated in Figure 1. Maximum use of *a priori* information is made in the design of the identification experiment, since this step may account for a significant portion of the identification cost. Previous work by Braun and coworkers (1999; 2000*a*; 2000*b*) has set forth guidelines to design minimum crest factor multisines and multilevel pseudo-random sequences for this purpose. Specifically, *a priori* information is used to design the input signal to contain power in the control-relevant frequencies, maintain identifiability of the nonlinear model from the data, and take into consideration time and input magnitude constraints. In the estimation/validation phase, the user verifies the input/output data covers the regressor spaces needed to reach the operational steady-states. Various user decisions can then be evaluated in a few iterations of Model-on-Demand validation. The final decisions are then incorporated in a Model-on-Demand Model Predictive Controller.

Model-on-Demand estimation is a nonparametric, nonlinear modeling approach conceptually similar to the Artificial Intelligence notion of Lazy Learning (Stenman, 1999*b*), but originally born out of the idea of Just-in-Time modeling (Cybenko, 1996). In contrast to global modeling techniques such as neural networks, semi-physical modeling, and block-oriented techniques, MoD adaptively selects the neighborhood size around the current operating point (i.e. bandwidth) which minimizes the bias/variance tradeoff. MoD does this by first selecting a range of candidate bandwidths and weighting the data according to the distance from the current operating point. Local models are

constructed from each candidate bandwidth, and the local model used corresponds to that which is optimal as determined by a localized classical bandwidth selector. Thus, the user is left with the following decisions which are well guided by traditional "linear thinking" (i.e. the modeling insights involved in identifying linear models):

- minimum and maximum data sizes to be considered for candidate bandwidth selection
- NARX regressor structure
- localized bandwidth selector

Other variations of the MoD formulation are possible, however the specific formulation discussed in Section 3 has demonstrated good performance at a low computational expense in most applications. By considering a nonlinear estimation technique which avoids the parameter initialization issue and iterative structural decisions, the user may focus instead on better design and analysis of the dynamical experiment and tuning of the MoDMPC controller.

In summary, MoD has the following advantages:

- the predictions made by the estimator are data-driven rather than structure dependent
- the predictions are less likely to get trapped by local minima in the optimization step as compared to global modeling
- new data can readily be added and/or old data can be discarded from the database
- the user is not required to determine good initial estimates of parameters
- regressor space partitioning associated with fuzzy-modeling and other local linear modeling techniques is avoided

The disadvantages of this approach include:

- brute-force database search for local subsets of data
- the estimate is dependent on on-line computational resources
- the estimator suffers from the "curse of dimensionality"
- very poor estimates may arise when the current operating condition falls outside of data support

Based on the current rate of advance of computer and data-handling technology, the first two disadvantages will become less relevant over time and MoD is well posed to take advantage of these new technologies. The last two items are common concerns affecting other black-box modeling techniques and are not unique to Model-on-Demand estimation.

3 Model-on-Demand Estimation and User Decisions

Suppose a noisy dataset $(Y_i, X_i)_{i=1}^N$ is available and some function m to construct a relationship

$$Y_i = m(X_i) + \epsilon_i, \qquad i = 1, ..., N, \tag{1}$$

is desired, where X_i represents an independent variable, Y_i represents a response variable, and ϵ_i depicts a set of independent, identically distributed random variables with zero means and known variances σ_i^2. m is a some nonlinear function that maps X_i to Y_i. N represents the number of observations in the database.

One way to estimate the true value y at the current operating point x is to take the average of the response variables produced by a small neighborhood of the independent variables around the current operating point. To try to improve the estimate, it is also possible to give more weight to data that are closer to the operating point and less weight to data that are farther away. The weights can be chosen according to a *kernel* function, which explicitly defines the shape of the weights much like the idea of windowing in digital signal processing. The shape of these windows and the amount of data that should reside in the neighborhood are issues which dramatically effect the performance of the estimate. The estimation of the response variable from a small portion of the dataset is known as local modeling.

In choosing a local model structure, it is desirable to use one that allows for a closed form, computationally efficient solution to the local estimation regression problem. By choosing a locally linear structure the simple least-squares solution is retained. Note that it is assumed that x and X_i can be represented in vector form (i.e. dynamic regressors) and have the same length. Using an l_2 norm

$$\hat{\beta} = \arg \min_{\beta} \sum_{i \in \Omega_k(x)} (Y_i - \mathcal{B}^T(X_i - x)\beta)^2 w_i(x), \tag{2}$$

where

$$\beta = (\beta_0 \quad \beta_1 \quad \dots \quad \beta_p)^T \tag{3}$$

β represents the parameter vector, where p is dependent on d the length of the X_i vector. For the linear case, p=d. For the quadratic case $p = d + d(d+1)/2$. $\mathcal{B}(X_i - x)$ represents the vector of regressors for the polynomial model. w_i represents the corresponding smoothing weight. $\Omega_k(x)$ denotes the selected regressor neighborhood. Hence, the locally linear model can be represented as

$$\mathcal{B}(X_i - x) = [1 \quad (X_i - x)^T]^T. \tag{4}$$

The MoD formulation can be naturally extended for MIMO prediction. Additional inputs, outputs, or measured disturbances can be included into $\mathcal{B}$ to improve the output prediction. Two MoD

estimators can be used in parallel to produce output predictions for two outputs, three for three outputs, and so on.

Tuning of the weights w_i takes place locally according to a multivariable kernel function of the form

$$w_i(x) = K\left(\frac{d(X_i, x)}{h}\right). \tag{5}$$

where K is a kernel function, d is a distance function and h is the bandwidth. The specific formulation in this work makes use of a tricube kernel function. This function is computationally efficient and has a continuous derivative. The function smoothly descends to zero near the outskirts. This property reduces high frequency noise leakage in the estimate as compared to the boxcar and other windows which do not smoothly descend to zero.

The distance function d is designed to determine how far away the data in $\Omega(k)$ are from the current operating point. This distance decides how the data will be weighted by the kernel function. This distance function can be weighted as

$$d(X_i, x) = \sqrt{(X_i - x)^T \mathbf{M} (X_i - x)}, \tag{6}$$

a fully scaled Euclidean distance function. Defining $\mathbf{M}$ to be equal to the inverse covariance of the regressors suits the formulation for this paper, since the inverse covariance of the regressors is readily available and provides an automatic scaling of the regressors to account for unit and value differences.

The bandwidth h that provides the best tradeoff between bias and variance errors in the estimate is not known *a priori*. The MoD concept begins by selecting enough data in the local neighborhood Ω_k, such that the least squares problem is not ill-conditioned. The neighborhood size is then expanded until a reasonable tradeoff between bias and variance errors is reached. Ideally, one seeks to minimize the point-wise Mean Square Error (MSE)

$$MSE(\hat{m}(x, h)) \triangleq E((\hat{m}(x, h) - m(x))^2, \tag{7}$$

which can be expressed as the sum of a squared bias term and a variance term,

$$MSE(\hat{m}(x, h)) = E(\hat{m}(x, h) - m(x))^2 + Var(\hat{m}(x, h)). \tag{8}$$

Recall that m is the true nonlinear relationship between the independent and dependent variables, while $\hat{m}$ is the estimated relationship. Observing the relationship between these two terms and the bandwidth h, one finds that the variance error of the estimate decreases, while the bias error of the estimate increases with increasing bandwidth. If the true function m was known, however, there would be no need to estimate it. Therefore, localized automatic bandwidth selectors are used to

estimate a goodness-of-fit. These selectors are the workhorses in MoD, striking a balance between bias and variance.

First, some notation will be defined so that the bandwidth selectors can be compactly presented. Let

$$\mathbf{y}_k \triangleq (Y_{i_1} \quad \dots \quad Y_{i_k})^T, \tag{9}$$

and define the local design matrix of basis functions as

$$\mathbf{X}_k \triangleq \begin{pmatrix} \mathcal{B}^T(X_{i_1} - x) \\ \vdots \\ \mathcal{B}^T(X_{i_k} - x) \end{pmatrix}. \tag{10}$$

The local weight matrix can be written

$$\mathbf{W}_k \triangleq diag(w_{i_1} \quad \dots \quad w_{i_k}). \tag{11}$$

The solution to the local regression problem and the corresponding estimate can now be written as

$$\hat{\beta} = (\mathbf{X}_k^T \mathbf{W}_k \mathbf{X}_k)^{-1} \mathbf{X}_k^T \mathbf{W}_k \mathbf{y}_k \tag{12}$$

$$\hat{m}(x,k) = [1 \quad 0 \quad \dots \quad 0]^T \hat{\beta}. \tag{13}$$

For notational rigor, also note that

$$\bar{m}(X_i,k) \triangleq \mathcal{B}^T(X_i - x)\hat{\beta}. \tag{14}$$

Localized AIC

The localized Akaike Information Criterion (AIC) is written below, where α represents the user adjustable penalty on the variance term. The AIC criteria has been a popular method for choosing the number of parameters in linear identification.

$$AIC(x,k) \triangleq \frac{\sum_{i \in \Omega_k(x)} w_i(x)(Y_i - \bar{m}(X_i,k))^2}{tr(\boldsymbol{W}_k)} exp(\frac{\alpha \; tr((\boldsymbol{X}_k^T \boldsymbol{W}_k \boldsymbol{X}_k)^{-1}(\boldsymbol{X}_k^T \boldsymbol{W}_k^2 \boldsymbol{X}_k))}{tr(\boldsymbol{W}_k)}) \tag{15}$$

Mallows C_p Criterion

If the system under study contains significant noise, a localized version of the Mallows C_p criterion can include an estimate of the noise variance. This also requires some modification of our estimate such that Equation 2 becomes

$$\hat{\beta} = \arg \min_{\beta} \sum_{i \in \Omega_k(x)} (Y_i - \mathcal{B}^T(X_i - x)\beta)^2 w_i(x) \cdot v_i. \tag{16}$$

Following in the matrix/vector notation, this is written

$$\hat{\beta} = (\mathbf{X}_k^T \mathbf{W}_k \mathbf{V}_k \mathbf{X}_k)^{-1} \mathbf{X}_k^T \mathbf{W}_k \mathbf{V}_k \mathbf{y}_k, \tag{17}$$

where

$$\mathbf{V}_k \triangleq diag(1/\sigma_{i_1}^2 \quad \dots \quad 1/\sigma_{i_k}^2). \tag{18}$$

The C_p criterion can now be written as

$$CP(x,k) \triangleq \frac{1}{tr(\mathbf{W}_k)} \Big(\sum_{i \in \Omega_k(x)} \frac{w_i(x)}{\sigma_i^2} (Y_i - \bar{m}(X_i,k))^2 \tag{19}$$
$$-tr(\mathbf{W}_k) + \alpha tr((\mathbf{X}_k^T \mathbf{V}_k \mathbf{W}_k \mathbf{X}_k)^{-1} \mathbf{X}_k^T \mathbf{W}_k^2 \mathbf{V}_k \mathbf{X}_k)).$$

The C_p criterion is preferable for a user faced with noisy systems since known variance information can be used to help improve the bandwidth selection.

Consider the selection of candidate values for bandwidth. To brute-force through the entire dataset, computing a model and evaluating the goodness-of-fit for the successive addition of one datapoint to the database would require an immense amount of time and computations. Instead, an exponential updating scheme is used as shown in the following procedure.

1. Fit at a very small bandwidth h_0, for which the estimation problem is well posed.
2. Increase bandwidth exponentially according to

$$h_i = C_h \cdot h_{i-1}, \quad C_h = 1 + \frac{0.3}{d} \tag{20}$$

   where $C_h > 1$, and d is the dimension of the regressor space. Compute the corresponding fits. Proceed until the goodness-of-fit falls below a predefined significance level.
3. Utilize the model of the bandwidth with lowest goodness-of-fit cost in 2.

In addition, the user can set a range on the number of data to be included in the local neighborhood k_{min} to k_{max}, in which the exponential update will search. Part of the validation of the MoD estimator involves picking appropriate values for this range. The user can plot the values of k used for each data point in the prediction and inspect the estimated output for instabilities due to poor data support. Based on this, the user has a good estimate of how far to increase k_{min}. In practice, there is no real limit on k_{max} other than computation time and database length. k_{max} can be set beyond the highest value for k_{opt} observed for the validation dataset.

Choosing k_{min} and k_{max} properly and determining a reasonable regressor structure n_a, n_b, and n_k, are the two critical decisions the user has to make. This paper supports the use of the best

fit global linear ARX model structure as the method of initial regressor selection for MoD. Very often the regressor space that provides the best fit for the global linear ARX modeling approach provides the best fit for the Model on Demand approach as well. This choice is attractive because the best regressor space for the linear case is readily available by using the classical crossvalidation techniques such as those found in the System Identification Toolbox for MATLAB^{TM} for both SISO and MISO identification. The global linear ARX model is in fact a special case of the MoD predictor. For example, by using a boxcar window instead of a tricube window and forcing the MoD predictor to use the entire estimation database and only the entire estimation database, a global linear ARX model would be created at each step. Therefore, regressors which work well for the global linear ARX case are likely to be a good starting point for the more general case of MoD estimation. Additional methods of regressor selection for nonlinear modeling are possible, such as the False Nearest Neighbors method of Rhodes and Morari (1998), and the MANOVA based analysis of Lind (2000).

The MoD user decisions can be summarized as follows:

1. Determine initial guesses of the NARX[n_a n_b n_c] structure via a global linear ARX model or by the False Nearest Neighbors method.

2. Select a localized bandwidth selector such as the AIC. For particularly noisy datasets, the localized Mallows C_p is an attractive choice if the variance of the noise in the data is known *a priori*.

3. Using a locally linear model ($p = 1$) and wide bandwidth range (something like $k_{min} = 15$ and k_{max} set equal to the length of estimation data), use MoD to produce an estimate of the output for the validation dataset.

4. Examine the MoD output and corresponding profile of k_{opt} to determine values of k where the MoD estimation is poor and/or the least squares solution was ill-conditioned. Set k_{min} above the highest value for which MoD performance was not desirable.

5. Re-estimate the output for the validation data. If performance is not desirable, repeat the previous step. For systems where high values of k still do not produce the desired performance, try a locally quadratic model ($p = 2$), or try the Mallows C_p criterion.

For most systems, the desired performance is achieved after step 4. For more difficult systems (or more difficult validation criteria), it may be necessary to try a range of NARX regressor structures that are near the global linear ARX structure. The validation is particularly user-friendly since the user is not left guessing at which terms to include in the model and whether poor performance is due to structure or initialization.

4 Model-on-Demand Model Predictive Control

To incorporate the MoD estimator into an MPC control framework, we start with the standard MPC objective function. Given the model description and knowledge of the current system state, the control objective is to minimize the function

$$J = \sum_{k=0}^{N_p-1} Q_e(k)(r(t+k+1) - \hat{y}(t+k+1))^2 + Q_u(k)u^2(t+k) + Q_{\Delta u}(k)\Delta u^2(t+k) \tag{21}$$

where $Q_e(k)$, $Q_u(k)$ and $Q_{\Delta u}(k)$ represent penalty weights on the control error, control signal, and control move size, respectively. r represents the setpoint trajectory, u the control signal, and $\hat{y}$ the estimated output.

The objective function 21, is solved using a receding horizon approach; of the N_u future control actions that minimize J, only the first one is used. Often only 4 or 5 control increments are solved for, with the implicit assumption that the control action will be held constant for the remaining moves up to the prediction horizon N_p. This is advantageous since optimization of Equation 21 can be computationally intensive for large prediction horizons N_p. The practice of using smaller control horizons N_u has the effect of producing less aggressive controllers and providing stable control for non-minimum phase systems (Meadows and Rawlings, 1997).

The main advantage of MPC is that hard constraints can be specified by the user on manipulated and controlled variable of the plant

$$u_{min} \leq u(t+k) \leq u_{min}, \tag{22}$$

$$\Delta u_{min} \leq \Delta u(t+k) \leq \Delta u_{min}, \tag{23}$$

and

$$y_{min} \leq y(t+k) \leq y_{min}. \tag{24}$$

In the standard GPC formulation, the plant model includes a disturbance term $v(t)$ which is assumed to represent random steps at random times, and these are not necessarily stationary (Clarke *et al.*, 1987).

$$A(q^{-1})y(t) = B(q^{-1})u(t) + v(t) \tag{25}$$

Based on these assumptions, $v(k)$ can be represented in the model as

$$v(k) = C(q^{-1})\xi(t)/\Delta \tag{26}$$

where $\xi(t)$ represents an uncorrelated random sequence, $C(q^{-1})$ represents a noise model and Δ represents the differencing operator $1 - q^{-1}$. Equations 25 and 26 are together referred to as the

Control Autoregressive with Integrator Moving Average model (CARIMA). The current formulation of the MoD estimator does not include an estimate of the noise model, so $C(q^{-1})$ is set to 1 and the assumptions about the disturbance require that the integrator be included in the plant model such that

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t). \tag{27}$$

To express the output prediction at time $t+k$ as a function of future controls, the Diophantine equations are solved

$$1 = A(q^{-1})F_k(q^{-1})\Delta + q^{-k}G_k(q^{-1}) \tag{28}$$

i.e.,

$$\hat{y}(t+k) = B(q^{-1})F_k(q^{-1})\Delta u(t+k-1) + G_k(q^{-1})y(t) \tag{29}$$

By partitioning $B(q^{-1})F_k(q^{-1})$ as

$$B(q^{-1})F_k(q^{-1}) = S_k(q^{-1}) + q^{-k}\tilde{S}_k(q^{-1}), \tag{30}$$

where deg $S_k(q^{-1}) = k-1$ and deg $\tilde{S}_k(q^{-1}) = n_b - 2$, the output prediction can be rewritten as

$$\hat{y}(t+k) = S_k(q^{-1})\Delta u(t+k-1) + \bar{y}(t+k). \tag{31}$$

The first term depends on future control actions whereas the remaining terms depend on measured quantities. By introducing

$$\begin{aligned}
\hat{\mathbf{y}} &\triangleq (\hat{y}(t+1) \quad \dots \quad \hat{y}(t+N_p))^T, \\
\boldsymbol{\Delta}\tilde{\mathbf{u}} &\triangleq \Delta u(t) \quad \dots \quad \Delta u(t+N_p-1))^T, \\
\bar{\mathbf{y}} &\triangleq (\bar{y}(t+1) \quad \dots \quad \bar{y}(t+N_p))^T, \\
\tilde{\mathbf{S}} &\triangleq \begin{pmatrix} s_0 & 0 & \dots & 0 \\ s_1 & s_0 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ s_{N_p-1} & s_{N_p-2} & \dots & s_0 \end{pmatrix},
\end{aligned}$$

where s_i are the coefficients of $S_k(q^{-1})$, thus the output prediction is

$$\hat{\mathbf{y}} = \bar{\mathbf{y}} + \tilde{\mathbf{S}}\boldsymbol{\Delta}\tilde{\mathbf{u}} \tag{32}$$

Making the assumption that the control horizon N_u is usually shorter than the prediction horizon N_p and the move sizes are assumed to be zero after N_u, then

$$\hat{\mathbf{y}} = \bar{\mathbf{y}} + \mathbf{S}\boldsymbol{\Delta}\mathbf{u}, \tag{33}$$

where

$$\boldsymbol{\Delta}\mathbf{u} \triangleq (\Delta u(t) \quad \dots \quad \Delta u(t+N_u-1))^T, \tag{34}$$

and

$$\mathbf{S} \triangleq \tilde{\mathbf{S}}\mathbf{\Lambda}, \quad \Lambda \triangleq \begin{pmatrix} I \\ 0 \end{pmatrix}. \tag{35}$$

The control vector can also be expressed

$$\mathbf{u} = \mathbf{T}\boldsymbol{\Delta}\mathbf{u} + \bar{\mathbf{u}} \tag{36}$$

where

$$\mathbf{T} \triangleq \begin{pmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{pmatrix}$$

and

$$\bar{\mathbf{u}} \triangleq \begin{pmatrix} u(t-1) \\ u(t-1) \\ \vdots \\ u(t-1) \end{pmatrix}.$$

This formulation will make use of m endpoint weights at the end of the prediction horizon, to invoke the stability properties discussed in (Demircioglu and Clarke, 1993). Thus, the dimensions of $\hat{\mathbf{y}}$, $\bar{\mathbf{y}}$, and $\mathbf{S}$, are increased to $(N_p + m)$, $(N_p + m)$, and $N_p + m \times N_p + m$, respectively. The objective can then be written in vector/matrix form

$$J = \| \boldsymbol{r} - \bar{\boldsymbol{y}} - \boldsymbol{S}\boldsymbol{\Delta}\boldsymbol{u} \|^2_{Q_e} + \| \boldsymbol{T}\boldsymbol{\Delta}\boldsymbol{u} + \bar{\boldsymbol{u}} \|^2_{Q_u} + \| \boldsymbol{\Delta}\boldsymbol{u} \|^2_{Q_{\Delta u}} \tag{37}$$

where

$$\boldsymbol{r} \triangleq (r(t+1) \ \dots \ r(t+N_p))^T \tag{38}$$

denotes the desired reference trajectory, and $\boldsymbol{Q_e}$, $\boldsymbol{Q_u}$, and $\boldsymbol{Q_{\Delta u}}$ are diagonal matrices with entries $Q_e(k)$, $Q_u(k)$, and $Q_{\Delta u}(k)$, respectively. Note the last m entries of $\boldsymbol{Q_e}$ correspond to the weightings for the endpoint constraints, and this sub-matrix will be denoted as $\boldsymbol{Q_w}$.

For the unconstrained case, there exists a closed form least squares solution to the MPC problem

$$\mathbf{u} = (\mathbf{S}^T\mathbf{Q_e}\mathbf{S} + \mathbf{T}^T\mathbf{Q_u}\mathbf{T} + \mathbf{Q_{\Delta u}})^{-1}(\mathbf{S}^T\mathbf{Q_e}(\mathbf{r} - \bar{\mathbf{y}}) + \mathbf{T}^T\mathbf{Q_u}\bar{\mathbf{u}}). \tag{39}$$

For the constrained case, a modification must be made to allow for handling of infeasible output constraint situations. Vectors of slack variables $\epsilon_{\mathbf{min}}$ and $\epsilon_{\mathbf{max}}$, penalize predicted output constraint violations at each instant along the prediction horizon. The problem can now be written

$$\min_{\Delta u, \epsilon} \| \boldsymbol{r} - \bar{\boldsymbol{y}} - \boldsymbol{S}\boldsymbol{\Delta}\boldsymbol{u} \|^2_{Q_e} + \| \boldsymbol{T}\boldsymbol{\Delta}\boldsymbol{u} + \bar{\boldsymbol{u}} \|^2_{Q_u} + \| \boldsymbol{\Delta}\boldsymbol{u} \|^2_{Q_{\Delta u}} + \| \epsilon \|^2_{Q_y} + Q_{y'}\epsilon \tag{40}$$

$$\text{s.t.} \qquad \mathbf{C}[\mathbf{\Delta u^T \Delta u^T \Delta u^T \Delta u^T}] \leq \mathbf{c} \tag{41}$$

$$\mathbf{S\Delta u} + \bar{\mathbf{y}} \leq \mathbf{y_{max}} + \boldsymbol{\epsilon}_{\mathbf{max}} \tag{42}$$

$$-\mathbf{S\Delta u} - \bar{\mathbf{y}} \leq -\mathbf{y_{min}} + \boldsymbol{\epsilon}_{\mathbf{min}} \tag{43}$$

where

$$\mathbf{C} \triangleq \begin{pmatrix} T \\ -T \\ I \\ -I \end{pmatrix},$$

$$\mathbf{c} \triangleq \begin{pmatrix} \Delta u_{max} - \bar{u} \\ -\Delta u_{min} + \bar{u} \\ \Delta u_{max} \\ -\Delta u_{min} \end{pmatrix},$$

and

$$\epsilon = \begin{pmatrix} \boldsymbol{\epsilon}_{\mathbf{min}} \\ \boldsymbol{\epsilon}_{\mathbf{max}} \end{pmatrix} \geq 0. \tag{44}$$

This problem can be expressed as a quadratic programming (QP) problem which can be efficiently solved using standard routines such as "qp.m" or "quadprog.m" subroutines in MATLAB.

The use of an endpoint weighting condition to provide a stability guarantee for GPC formulations was first proposed by Demircioglu and Clarke (1993). By choosing a weight for Q_w beyond some value γ, some guarantees regarding stable control action and stabilization of unstable systems can be made. γ is dependent on the system under control, N_p, N_u, and $Q_{\Delta u}$. However, using a high weight may also provide control action which is too aggressive. For systems which exhibit inverse response, a high penalty may in fact lead to unnecessarily large inverse responses when step setpoint trajectories are followed. Thus the user may need to adjust Q_w to improve control performance. Note that this endpoint condition can provide stable control, provided $N_p \geq n+1$, and $m \geq n+1$, where n is the order of the system.

The output constraint infeasibility handling is adopted from Scokaert and Rawlings (1999). Their approach has shown to provide more intuitive output constraint infeasibility handling as opposed to penalizing output constraints along the entire prediction horizon with only one slack variable. The slack variables are penalized with both an ℓ_1-norm and ℓ_2-norm, allowing a better conditioned problem, and an extra degree of freedom for tuning. Large values of Q_y and $Q_{y'}$ have the effect of reducing the magnitude of the constraint violation at the expense of a longer duration of violation.

Smaller values of Q_y and $Q_{y'}$ allow the output violation magnitude to be large, but with short duration. Scokaert and Rawlings point out that while their results are developed for infinite horizon formulations, the results should hold for large N_p .

This formulation of MoDMPC chooses to use ideas of linear and adaptive MPC to maximize the use of insights gained from linear MPC approaches in addressing the nonlinear process control problem. The formulation is sufficient to provide proof-of-concept results, and can be considered a candidate control scheme for process systems that do not exhibit pathological properties. Formulation of MPC/GPC controllers addressing output constraint infeasibilities, stable control of nonlinear systems, disadvantages of qp solution methods and the similar issues are still active areas of research (Mayne *et al.*, 2000; Kouvaritakis *et al.*, 1997; Ebert *et al.*, 1998).

5 Case Studies: Model-on-Demand Model Predictive Control Performance

5.1 Non-adiabatic Continuous Stirred Tank Reactor

The CSTR model is based on a first principles analysis of a hypothetical CSTR (Denn, 1986; Bequette, 1998). The vessel is assumed to be perfectly mixed, and a single first-order exothermic, irreversible reaction, (A $\Rightarrow$ B), takes place. A diagram showing the vessel and the surrounding cooling jacket is presented in Figure 2. Based on accounting and conservation principles, the lumped parameter equations describing the system are

$$\begin{aligned} \frac{dC_A}{dt} &= \frac{F}{V}(C_{A_f} - C_A) - r \\ \frac{dT}{dt} &= \frac{F}{V}(T_f - T) - \left(\frac{\Delta H}{c_p \rho}\right) r - \frac{UA}{c_p \rho V}(T - T_j) \\ r &= k_o exp\left(\frac{-E}{RT}\right) C_A \end{aligned}$$

These case studies are concerned with the Case 2 reactor parameter values presented by Bequette and shown in Table 1. The inlet stream is fed at a constant rate F with constant concentration C_{Af} into the vessel. The final concentration of the reactant C_A is the controlled variable and the jacket temperature T_j is manipulated to keep the exit stream concentration C_A at setpoint. The exiting stream leaves at a rate F and since it is assumed the vessel is perfectly mixed, the exiting concentration and vessel concentration are assumed to be the same. The equations have been solved in a Matlab^{TM} Simulink environment, using a fixed-step size $T = 0.1\ hrs$ and a 4^{th} order Runge-Kutta solver.

5.1.1 Model-on-Demand Model Predictive Control vs. Linear MPC

The goal is to develop both a nonlinear MoDMPC controller and a linear MPC controller and contrast their performance for setpoint tracking about the high concentration, low temperature steady-state. To achieve this end, a multi-level PRS signal was designed for excitation of the CSTR for use in the MoDMPC controller. Signal designs of length beyond one day's time were considered unacceptable. Previous attempts to formulate a linear PID controller revealed that a high bandwidth controller was needed to keep the reactor from "igniting." Therefore, an α of 5 and a β of 1 were selected. A series of step tests revealed a dominant time constant range between 1.3 and 2 hours for this reactor. A sampling time of 0.1 hrs was used for the multi-level PRS and output measurements. For a switching time T_{sw} of 0.7 hrs (which met the high frequency limit), only a 3-level PRS signal generated with 3 shift registers or a 5-level PRS signal generated with 2 shift registers meets the requirements. The 5-level is the signal of choice because 5 levels offer better resolution of the input space than does 3. Furthermore, if one considers the Taylor expansion of the exponential in the kinetic equation, the highest possible number of input levels should be used. Tables 2 and 3 provide an example of how the choices for shift register and number of elements in the Galois field can be evaluated using spreadsheet calculations. The maximum move size of this signal is 50 K, the signal spans between 273.15 and 323.15 K, and the length was 16.8 hrs. The input range is particularly interesting because the diabatic CSTR exhibits an "ignition" as the reactor settles from a +9 K or greater increase in jacket temperature. Note there is an assumption of both heating and chilling capacity being available for the CSTR jacket and the achievable jacket temperature is constrained between 275 and 375 K.

The validation input consisted of a swept sine wave and random steps designed to test the interpolative ability of the MoD estimator and reach the same output spaces as the 5-level PRS experiment. The validation input has similar frequency content as that of the 5-level PRS signal. The input/output data and the validation results from the identification experiment are shown in Figure 3.

The linear MPC model comes from fitting an ARX[221] model to the 5-level PRS data. The ARX structure had the best fit on the validation data and was determined through brute-force selection by examining $1 \leq n_a \leq 10$, $1 \leq n_b \leq 10$, and $1 \leq n_k \leq 10$. The MoD estimator also made use of a NARX[221] structure, and validation proved reasonable user decisions to be $k_{min} = 55$ and k_{max} to be the length of the estimation database. A localized AIC criteria was used with a variance penalty of 3. Note from Figure 3, the MoD estimator is able to make better use of the nonlinear information contained in the 5-level PRS input/output data. As shown in Figure 4, the validation input/output data falls in the same spaces covered by the 5-level PRS. Note also, because of the adaptive nature of the MoD algorithm, the locally linear models on the outskirts of the input/output spaces provide better estimation of the plant response in those areas, than does one linear model fit to the entire

estimation dataset.

The control evaluation focuses on setpoint tracking ability of both the linear and MoDMPC controllers. The results with and without input constraints ($275 \leq u \leq 375$), are shown in Figures 5 and 6, respectively. The control parameters are the same for both controllers and are shown in the figures. The results show the advantage of nonlinear identification and control for this CSTR. Not only does the MoDMPC controller force the process to setpoint faster, it makes better use of control energy to do so. This is due to the more accurate predictions of MoD estimation as compared to the global linear ARX model. The slight overshoot in the MoDMPC results might further be reduced by considering the "linearizations along a trajectory" approach discussed in Nevistić (1997).

5.1.2 Control Utilizing Measured Disturbance Information

To provide a challenging study of the benefits of measured disturbance information to MoDMPC control, the CSTR simulation is modified to include a 0.4 hour time delay between the effect of a change in jacket temperature on the outlet concentration. The inlet concentration measurement is made available to the MoDMPC controller 0.7 hours before it will effect the outlet concentration. During the control experiment the inlet concentration will exhibit a low frequency drift with a span between 9.9 and 10.1 $kgmol/m^3$. Two MoDMPC controllers are designed for the system. For the first MoDMPC controller, the inlet concentration is held constant for the identification experiment. The identification experiment for the second MoDMPC controller includes the drifting inlet concentration as a measured disturbance.

The input design was modified from the previous section by considering more low frequency information, since more data is needed to characterize the effect of the low frequency drift on the outlet concentration. A 7-level PRS signal was used which produces a signal with low frequency content (0.19 rad/hr) corresponding to just slightly less than the 95% settling time of the process (0.17 rad/hr). The 7-level PRS, with 0.7 hr switching time, has a length of 33.6 hrs. Both identification experiments make use of the same 7-level PRS input and the validation input from the previous section. The MoD user decisions used in the previous section allowed the MoD estimation to perform well, and the NARX structures were modified to account for the transport delays added to the simulation.

In Figure 7, observe that with the same controller parameters, the MoDMPC controller with measured disturbance information provides better disturbance rejection. Both controllers satisfied input constraints. The RMS error between the setpoint and controlled variable was 0.413 for the MoDMPC controller that did not use the measured disturbance information. The MoDMPC controller utilizing the measured disturbance information had an RMS error of 0.392. Moreover,

by utilizing measured disturbance information 39% less valve movement is observed. The control performance improvement is most notable during the time ranges of 12 to 15 hours and 19 to 25 hours.

Inspecting the manipulated variable performance is critical in process applications, since it has a direct impact on actuator maintenance costs. Additional considerations include the impact on energy integration and the possible creation of disturbances to feed or exit streams of other process units in the plant. The CSTR sees the most benefit from judicious use of the manipulated variable by the MoDMPC controller with measured disturbance information. By looking at the sum of moves made throughout the control experiment (which we will refer to as total amount of movement or TAM), a quantitative comparison can be made. Specifically, the MoDMPC controller utilizing the measured disturbance information uses significantly less total amount of movement $TAM = 92.68\ K$ to achieve a higher level of performance than the standard MoDMPC controller $TAM = 150.68\ K$.

5.1.3 Control at an Unstable Steady-State

The ability of a MoDMPC controller to stabilize an unstable system is demonstrated by considering the intermediate steady-state conditions discussed by Bequette ($C_A = 5.51\ kgmol/m^2$, $T = 339.1\ K$). A closed-loop identification strategy was employed using the 5-level PRS and validation inputs previously presented, as the setpoint trajectory for a proportional-only controller. The controller gain $K_c = -9.5$ was tuned to provide stable performance for the identification experiment.

A MoDMPC controller is initially commissioned with a move suppression penalty which is too large $Q_{\Delta u} = 0.5$ to provide stable control. By adding three endpoint penalties to the end of the prediction horizon, the speed of response of the controller is increased, such that the process is stabilized. In Figure 8, the effect of increasing the endpoint penalties for various values up to 10^{10} is demonstrated. For $Q_w = 10^{10}$, the control action is most aggressive, repeatedly forcing the manipulated variable to constraints.

As mentioned in Demircioglu and Clarke (1993) an endpoint constraint is only a sufficient condition for stability, and endpoint constraints (or infinite weight endpoint conditions) may have an adverse effect on controller performance. The example given by Clarke was an unstable, non-minimum phase linear system, which exhibited inverse response. The example showed that the large values of Q_w has the effect of increasing the magnitude of the inverse response when the controller executes the requested step change. Thus, choosing values for Q_w becomes more difficult when controlling non-minimum phase nonlinear systems in the vicinity of output constraints, as demonstrated in the next section.

5.2 Control of a Nonlinear Non-minimum Phase System with Output Constraints

A nonlinear non-minimum phase system was developed to evaluate the ability of MoDMPC to handle output constraint infeasibilities, assess interaction of MoDMPC tuning parameters, and demonstrate performance. A discrete Hammerstein block model (shown in Figure 9), was developed to exhibit inverse response, and a static second order nonlinear dependence on the input.

The simulation was excited with a uniform random number sequence of $10,000$ datapoints, providing sufficient coverage throughout the expected operating region of the input/output space. The excitation used was based on theoretical arguments, to provide optimal excitation of the nonlinear system (Leontaritis and Billings, 1987). Thus, performance and stability can be assessed under theoretically ideal conditions and any degradation in performance cannot be attributed to lack of data support. Half of the data was used for estimation and the other half was used for validation. Through validation, reasonable user decisions proved to be an AIC localized bandwidth selector, a variance penalty of 3, $k_{min} = 30$, and $k_{max} = 200$.

To investigate the extent of interactions between the output constraint infeasibility penalties and the move suppression penalty, control experiments were designed to test the ability of the controller to follow a setpoint trajectory that switches between 10 and 32, while maintaining output constraints of 9 and 33. Additional control parameters that were kept constant throughout the experiments are shown in Table 4. By inspection of Figure 10, the results demonstrate that *both* the output constraint penalties, and move suppression penalty have the effect of slowing controller action for this problem as they are increased. It is also evident that as the move suppression and output constraint violation have a small penalty, the system becomes oscillatory, since the controller alternates between reducing the cost of meeting setpoint and reducing the cost of violating output constraints.

As noted in Section 3, the endpoint condition used to guarantee stability has the effect of increasing the closed-loop speed of response as the penalty is increased. In Figure 11, a comparison of increasing values for Q_w is made, that demonstrates the oscillatory phenomena observed in Figure 10. Thus, as setpoint trajectories take the process closer to operating constraints, the speed of response (i.e. the move suppression $Q_{\Delta u}$ and endpoint condition Q_w) of the controller directly impacts performance.

5.3 Control of a MIMO Rapid Thermal Processing (RTP) Reactor

The ability to handle a MIMO process is a necessity for any proposed MPC controller. To show proof-of-concept of MoD's ability to handle MIMO estimation and control, this section considers

control of a 6×6 RTP wafer reactor simulation. Honeywell Technology Center has developed a simplified model of a RTP wafer reactor, (Christoffel *et al.*, 1995). The reactor consists of five halogen/tungsten lamps placed concentrically on the floor of the reactor. Above the lamps is a quartz window. The wafer to be processed is positioned above the quartz window. The reacting gases are pumped into the reactor through the gas shower head in the top of the reactor. Figure 12 shows a schematic of the reactor configuration. The 6 inputs to the system are the percent power to each lamp, ranging from 0 to 100%. The 6 outputs of the system are the temperatures [K] of the wafer directly above the lamps.

The simplified model developed by Honeywell is

$$\hat{y}_k = P_1 \cdot y_{k-1} + P_2 \cdot (\frac{y_{k-1}}{1000})^4 + P_3 \cdot u_{k-1} + P_4. \tag{45}$$

This model, with the matrices P_i identified by Honeywell, served as the 'truth model' in our study. The quartic terms were included based on the assumption of heat transfer gains and losses due to radiation.

This simulation runs at 20 Hz and the data was resampled at a rate of one sample every 1/4 of a second. This system displays a dominant time constant range between 6 and 32 seconds. For guidelines for the design of MIMO multi-level PRS excitation signals the reader is referred to (Braun and Rivera, 2000). α was chosen to be 4 and β to be 3. These choices left many feasible multi-level PRS signals to choose from. A signal length beyond 4 hours was considered to long for the identification. The MIMO length requirement determined that a signal less than 0.317 hours was too short to provide uncorrelated signals for 6 inputs. A 17-level PRS generated with 2 shift registers, a switching time $T_{sw} = 4\ sec$ and even harmonic suppression was used to excite lamp power #1. Five versions of the same signal, delayed by increasing multiples of the 95% settling time excited the remaining channels. Delaying the excitation allows the multi-level PRS inputs to remain uncorrelated up to the 95% settling time. The input amplitudes for all 6 of these signals were between 0 and 100% lamp power.

The resulting input/output data did not cover the entire input/output trajectory which follows the setpoint trajectory, however it did provide support in the higher temperature region (1200 to 1400 K) where it is critical for the controller to level off wafer temperature, while maintaining temperature uniformity across the wafer. For the other regions, the controller must rely on the local models on the boundaries of support and the redeeming characteristics of feedback. A validation sequence consisting of a series of steps in the high temperature region was constructed and used to validate the MoD user decisions. An example of how the input/output data are represented in the time domain and various input/output spaces is demonstrated for lamp # 1 and temperature # 1 and their combination with lamp #2 and temperature # 2 in Figures 13 and 14. An AIC criteria with a

variance penalty of 3, $k_{min} = 300$, and $k_{max} = 1210$ provided good estimation of the validation data. The NARX structure predicted each output based on one past value of each output and each input.

In Figure 15, the control results for a 50 K/s ramp rate temperature trajectory are shown. Note the setpoint trajectory was supplied to the controller to provide anticipative action. No endpoint weight was used, and the prediction horizon was purposefully kept short to prevent the controller from making control moves too far in advance of the setpoint change. The controller parameters can also be found in Figure 15. The MoDMPC controller is able to keep the temperature across the wafer diameter to within ± 0.5 K.

6 Conclusions

The methodology provided in this paper allows the user to proceed through a framework of decision making, which can reduce the length and cost of the nonlinear identification and control design process. These benefits are possible through maximum use of *a priori* information, and by the use of the Model-on-Demand estimation technique which circumvents the initialization and structural decisions of more time consuming methods such as neural networks and semi-physical modeling.

By incorporating an endpoint condition for stability, and effective output constraint infeasibility handling, MoDMPC is positioned to effectively handle traditional nonlinear process systems and typical operating conditions (e.g. operating near constraints, and utilizing measured disturbance information). In addition, MoDMPC can outperform linear MPC by making better use of control action and keeping the output closer to setpoint by taking into account nonlinear effects. These performance benefits have been demonstrated on a SISO Continuous Stirred Tank Reactor (CSTR), and MIMO Rapid Thermal Processing (RTP) reactor. Tuning issues as they relate to operation near output constraints were addressed with a simulated Hammerstein system, exhibiting a 2^{nd}-order static nonlinearity and inverse response dynamics.

7 Software

The authors have developed a Model-on-Demand Model Predictive Control toolbox for MATLAB^{TM}. More information about the package and download instructions can be found on the internet at

`http://www.eas.asu.edu/~csel/software.htm`

The package contains a graphical user interface for data visualization and validation, as well as a Simulink^{TM} library of MoDMPC controllers.

8 Acknowledgments

The authors would like to thank Wendy Foslien of Honeywell Technology Center for providing the RTP reactor simulation, and Professor H. Anthony Barker of the University of Wales Swansea, UK for the use of his *Galois* Software which conveniently generates multi-level pseudo random sequences. Funding for this work was graciously provided by an NSF Graduate Research Traineeship (NSF ERH-9355079) and the Dupont Educational Aid Program.

References

Bequette, B. W. (1998). *Process Dynamics: Modeling, Analysis, and Simulation*. Prentice Hall PTR. Upper Saddle River, NJ.

Braun, M. W. and D. E. Rivera (2000). Design of multi-level pseudo-random sequences for control-oriented identification of nonlinear process systems. *Submitted for Publication*.

Braun, M. W., B. McNamara, D. E. Rivera and A. Stenman (2000*a*). "model-on-demand" identification for control: An experimental study and feasibility analysis for mod-based predictive control. In: *Proc. of IFAC SYSID2000*. Santa Barbara, CA.

Braun, M.W., D.E. Rivera, A. Stenman, W. Foslien and C. Hrenya (1999). Multi-level pseudo-random signal design and "model-on-demand" estimation applied to nonlinear identification of a rtp wafer reactor. In: *Proc. of the American Control Conf., San Diego, CA, USA, 2-4 June 1999*. pp. 1573–1577.

Braun, M.W., R. Ortiz-Mojica and D.E. Rivera (2000*b*). Design of minimal crest factor multisinusoidal signals for "plant-friendly" identification of nonlinear process systems. In: *Proceedings of the l2th IFAC Symposium on System Identification, Santa Barbara, CA, USA, 2-4 June 2000*.

Christoffel, J., W. Foslien, A. Mathur and M. Ekblad (1995). Model based control and optimization techniques from the advanced materials industry applied to semiconductor processes. In: *AEC/APC Workshop VII*.

Clarke, D. W., C. Mohtadi and P. S. Tuffs (1987). Generalized predictive control – part i. the basic algorithm. *Automatica* **23**(2), 137–148.

Cybenko, G. (1996). Just-in-time learning and estimation. In: *Identification, Adaptation, Learning* (S. Bittani and G. Picci, Eds.). pp. 423–434. NATO ASI. Springer.

Demircioglu, H. and D. W. Clarke (1993). Generalised predictive control with end-point state weighting. *IEE Proceedings-D Control Theory and Applications* **140**(4), 275–292.

Denn, M. M. (1986). *Process Modeling*. Longman. New York.

Ebert, W., R. Röstel, B. Meffert, W. Weller, F. Winkler, M. Günther, L. Heese, O. Hochmuth and G. Knell (1998). Multivariable end-point weighted generalized predictive control of a space crystal furnace. In: *Proceedings of the 1998 IEEE: International Conference on Control Applications*. Trieste, Italy. pp. 527–531.

Hussain, M. A. (1999). Review of the applications of neural networks in chemical process control – simulation and online implementation. *Artificial Intelligence in Engineering* **13**(1), 55–68.

Kouvaritakis, B., J. A. Rossiter and J. R. Gossner (1997). Improved algorithm for multivariable stable generalised predictive control. *IEE Proceedings - Control Theory and Applications* **144**(4), 309–312.

Leontaritis, I. J. and S. A. Billings (1987). Experimental design and identifiability for non-linear systems. *International Journal of System Science* **18**(1), 189–202.

Lind, I. (2000). Model order selection of n-fir models by the analysis of variance method. In: *Proc IFAC Symposium SYSID 2000*. Santa Barbara. pp. We PM4–4.

Mayne, D. Q., J. B. Rawlings, C. V. Rao and P. O. M. Scokaert (2000). Constrained model predictive control: Stability and optimality. *Automatica* **36**, 789–814.

Meadows, E. S. and J. B. Rawlings (1997). *Nonlinear Process Control*. Chap. Model Predictive Control, pp. 233–310. Prentice Hall PTR. Upper Saddle River, NJ.

Nevistić, V. (1997). Constrained Control of Nonlinear Systems. PhD thesis. Swiss Federal Institute of Technology (ETH). Zurich, Switzerland.

Rhodes, C. and M. Morari (1998). Determining the model order of nonlinear input/output systems. *AIChE Journal* **44**(1), 151–163.

Scokaert, P. O. M. and J. B. Rawlings (1999). Feasibility issues in linear model predictive control. *AIChE Journal* **45**(8), 1649–1659.

Stenman, A. (1999*a*). Model-free predictive control. In: *Proceedings of the 38th IEEE Conference on Decision and Control, Phoenix, AZ, USA*. pp. 3712–3717.

Stenman, A. (1999*b*). Model on Demand: Algoritms, Analysis and Applications. PhD thesis. Linköping University. SE-581 83 Linköping, Sweden.

Table 1: Non-adiabatic CSTR, Case 2 reactor parameters

Parameter	Units	Value
F/V	hr^{-1}	1
k_0	hr^{-1}	9703*3600
$(-\Delta H)$	kcal/kgmol	5960
E	kcal/kgmol	11843
$c_p\rho$	kcal/($\text{m}^3{}^o\text{C}$)	500
T_f	^{o}C	25
C_{Af}	kgmol/m^3	10
UA/V	kcal/($\text{m}^3{}^o\text{Chr}$)	150
T_j	^{o}C	25

Table 2: Selection of multi-level PRS design variables to meet low frequency limit $\omega_* = 0.166$

low frequency rad/hr	Elements q		
shift register n_r	3	5	7
2	1.122	0.374	0.187
3	0.345	0.072	0.026
4	0.112	0.014	0.004

Table 3: Selection of multi-level PRS design variables to meet length limit $T_{cyc} \leq 1$ day

length hr	Elements q		
shift register n_r	3	5	7
2	5.6	16.8	33.6
3	18.2	86.8	239.4
4	56.0	436.8	1680.0

Table 4: Controller parameters for control of the non-minimum phase nonlinear block system

Control Parameter	Symbol	Value
Tracking Error Weight	Q_e	1
Control Weight	Q_u	0
Endpoint Condition Weight	Q_w	0
Number of Endpoints	m	0
Prediction Horizon	N_p	18
Control Horizon	N_u	3
Input Constraints	$u_{min},\ u_{max}$	1, 5

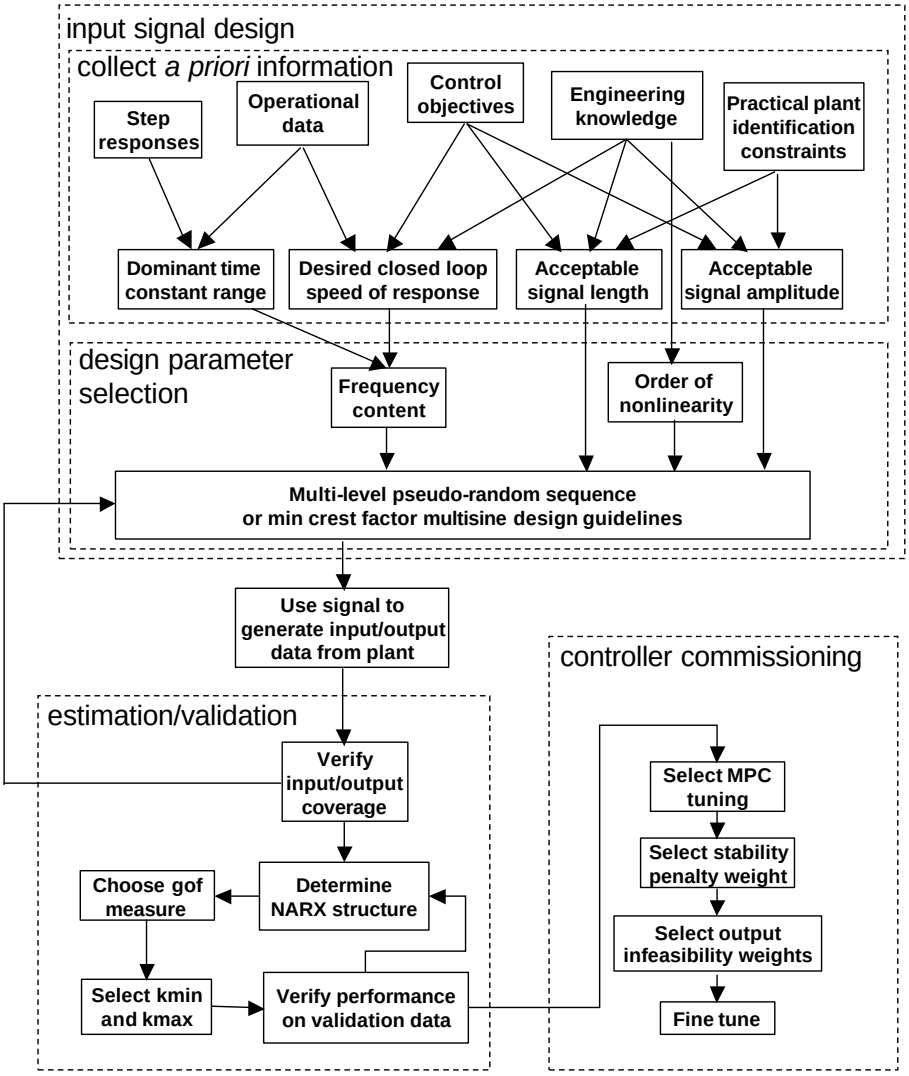

Figure 1: Model-on-Demand Identification and Control Methodology

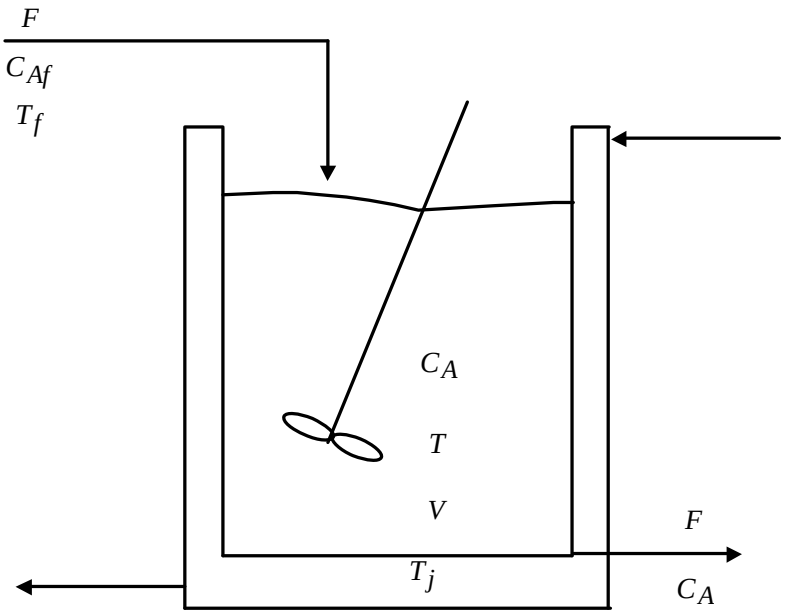

Figure 2: Schematic of the diabatic CSTR

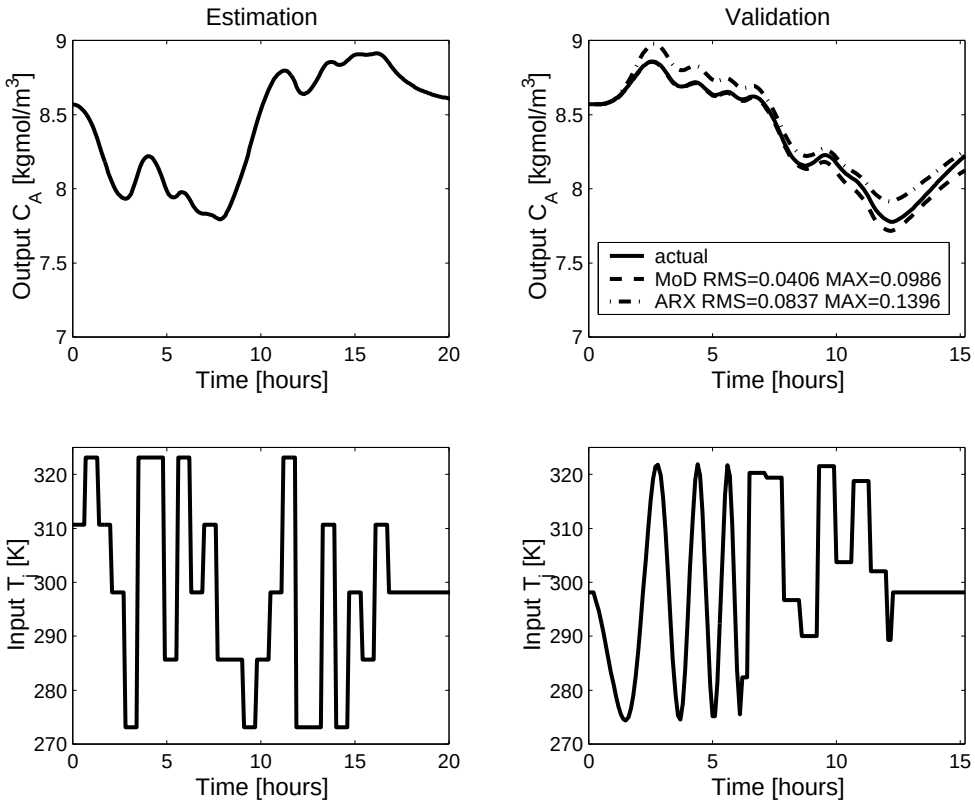

Figure 3: Estimation and Validation Data for CSTR case study

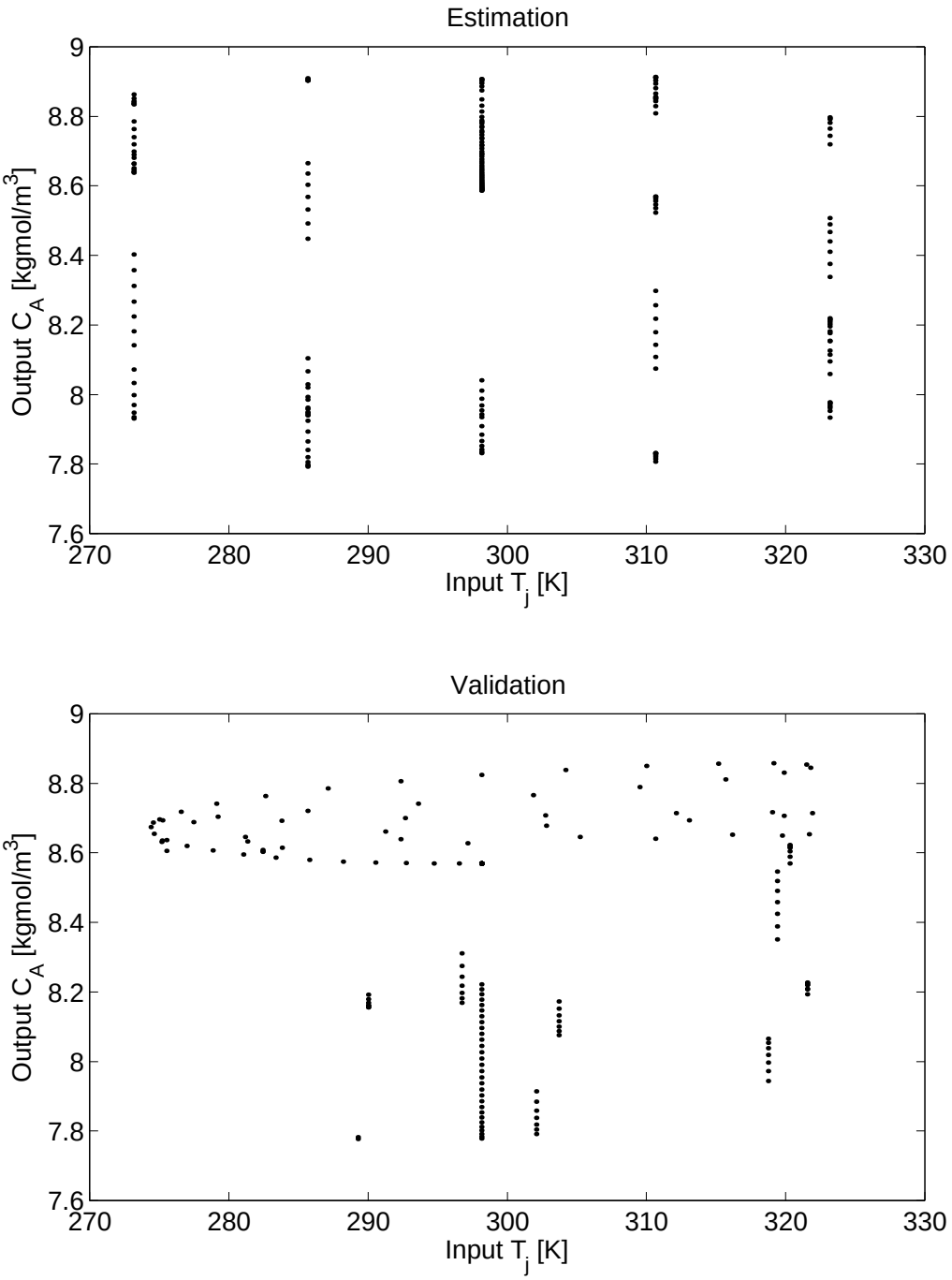

Figure 4: Input/output spaces of CSTR datasets

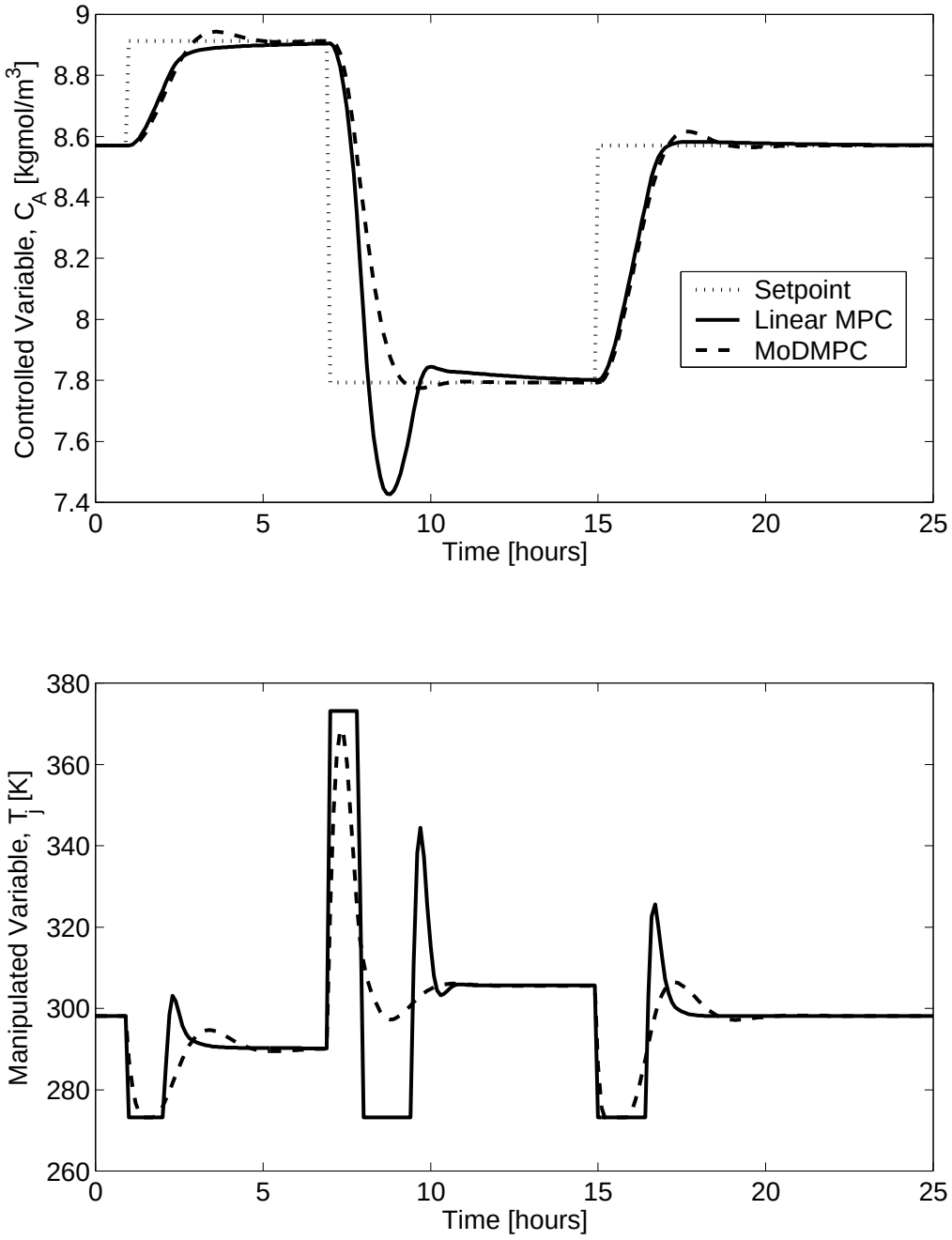

Figure 5: Control with input constraints; $Q_e = 1,\ Q_u = 0,\ Q_{\Delta u} = 0.001,\ N_p = 14,\ N_u = 3$

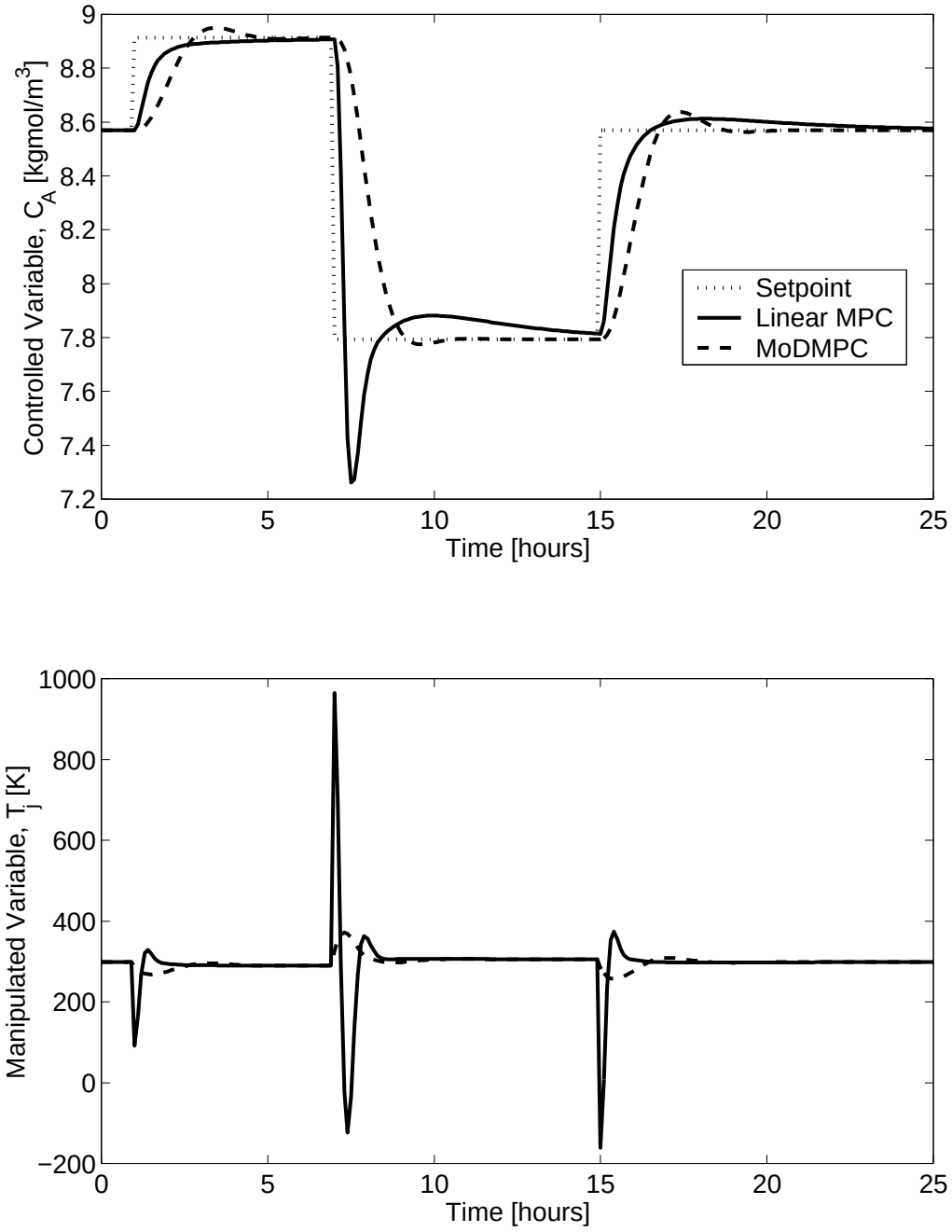

Figure 6: Control without input constraints; $Q_e = 1,\ Q_u = 0,\ Q_{\Delta u} = 0.001,\ N_p = 14,\ N_u = 3$

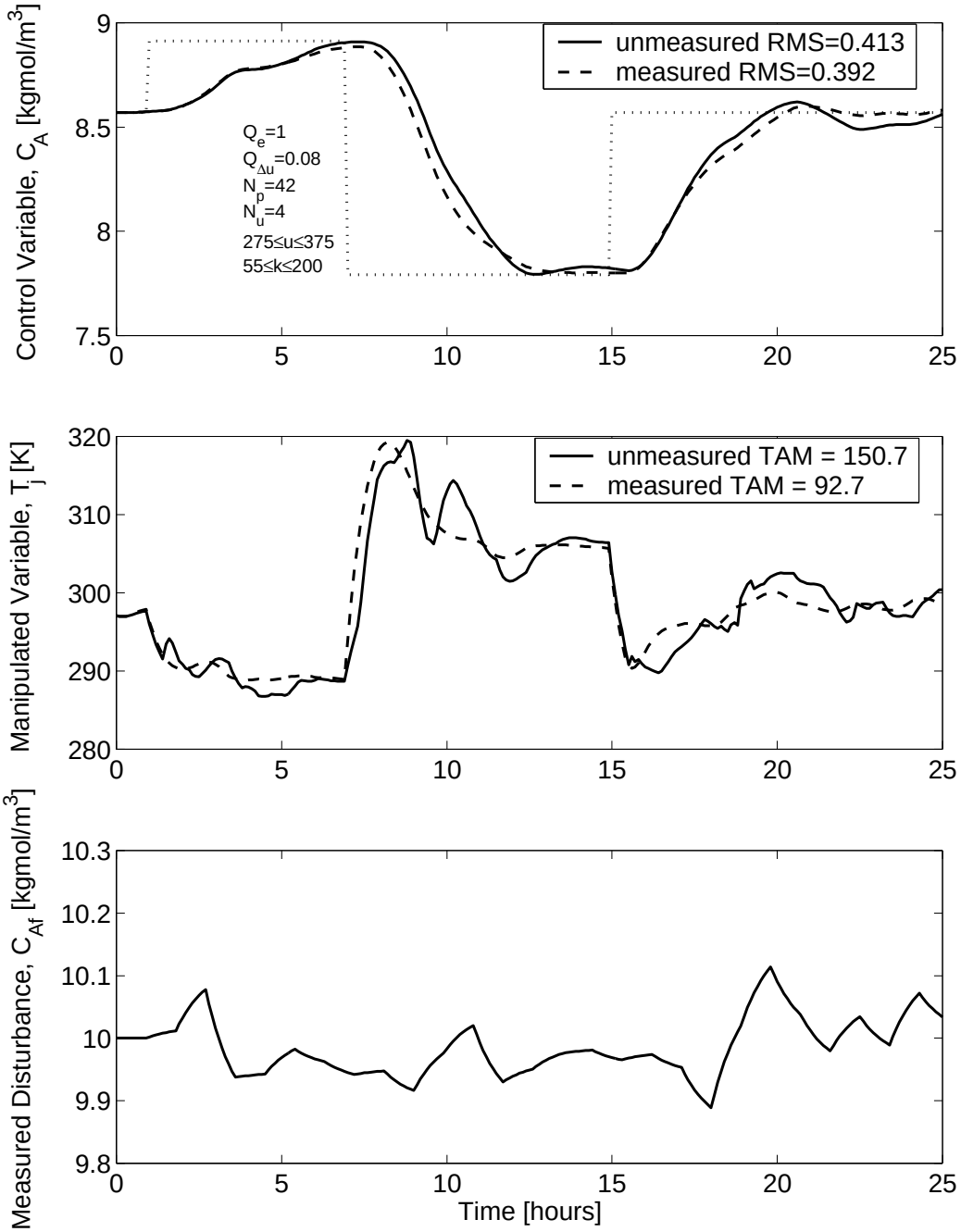

Figure 7: Comparison of MoDMPC with and without utilizing measured disturbance information

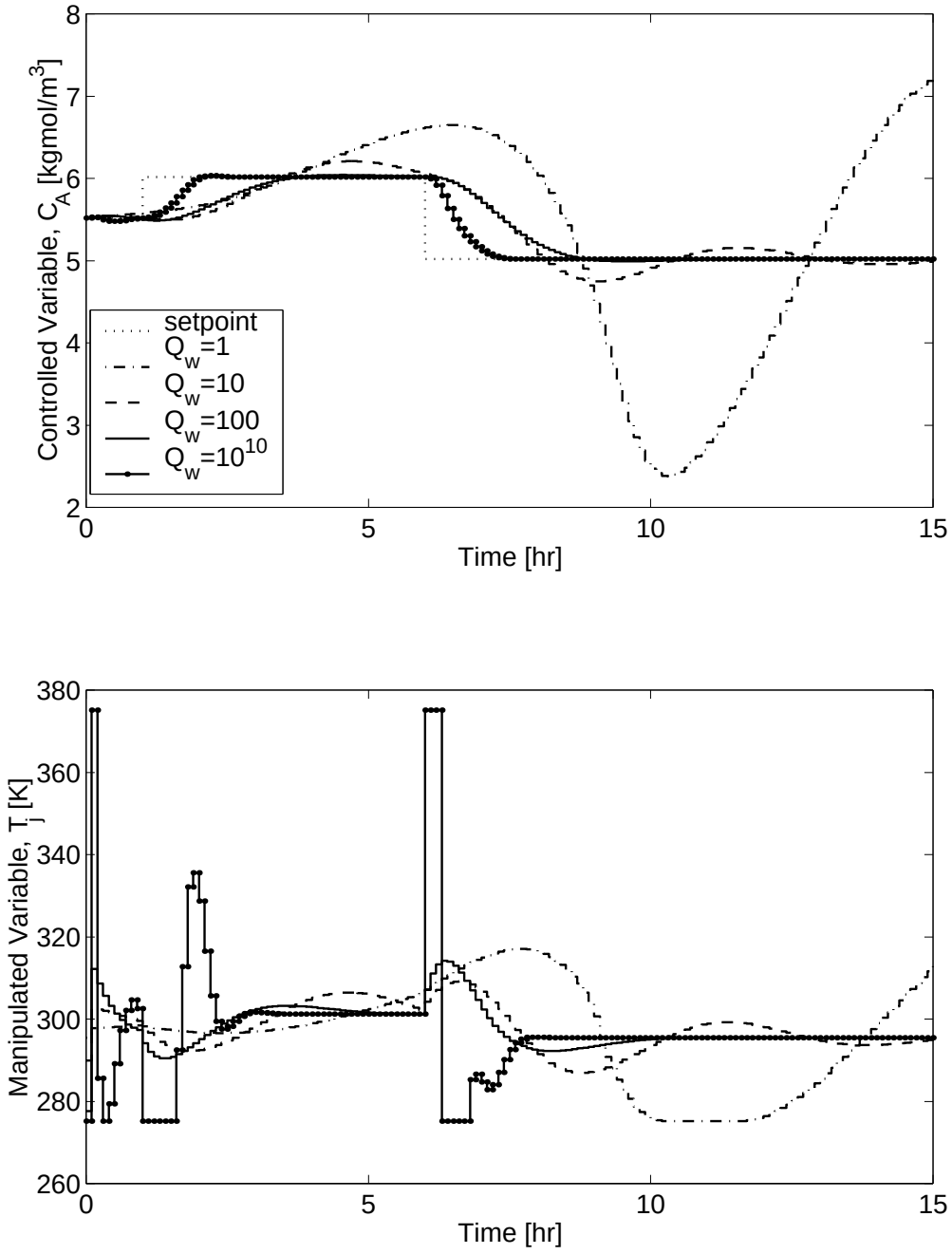

Figure 8: MoDMPC control about the CSTR unstable steady state

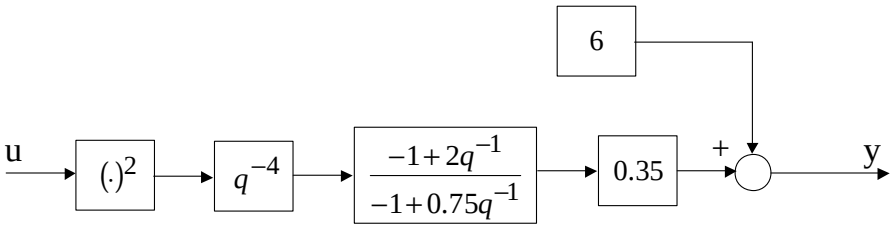

Figure 9: Block diagram of Hammerstein model

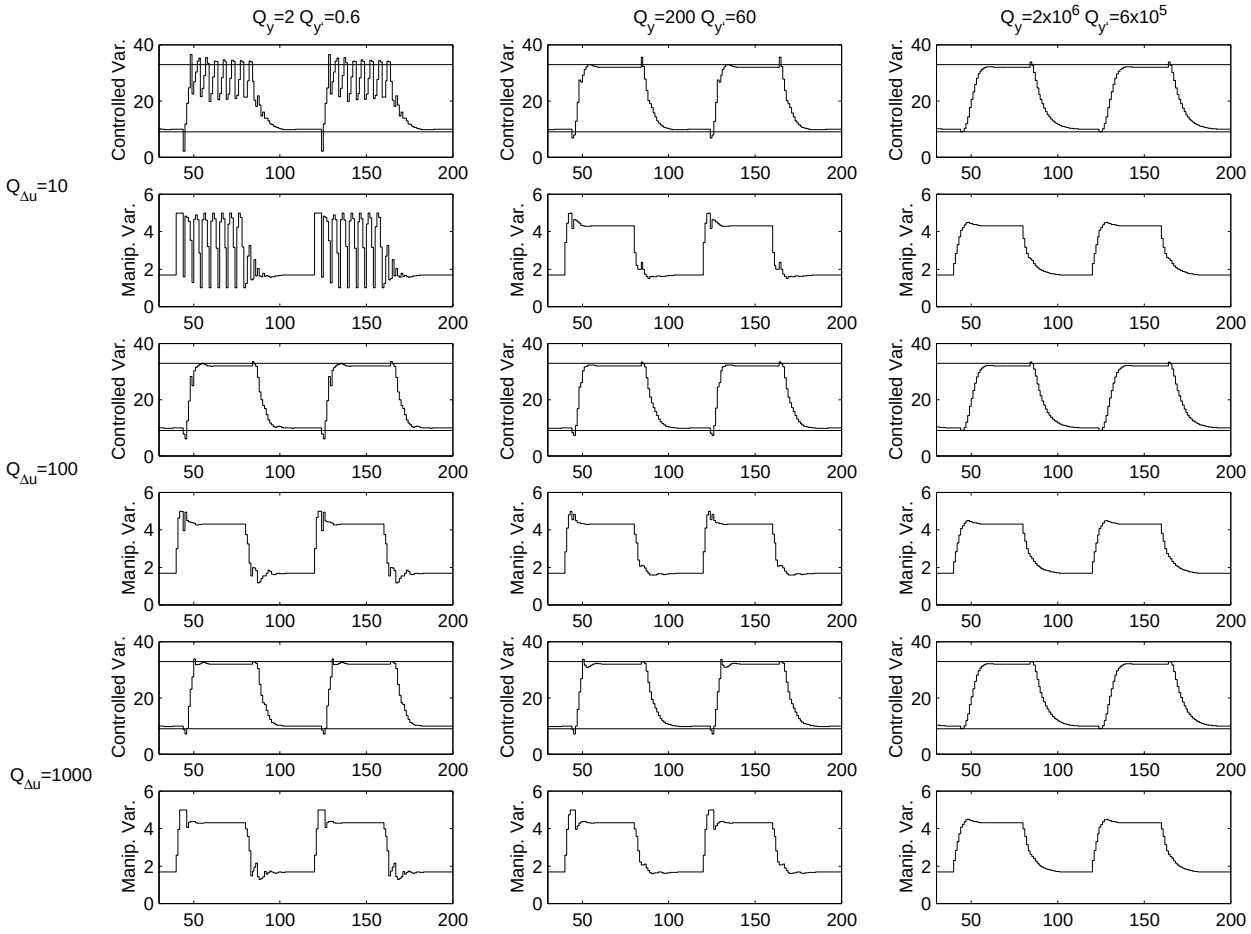

Figure 10: Effects of tuning parameters $Q_{\Delta u}, Q_y$ and $Q_{y'}$ on MoDMPC control performance in the presence of output constraints

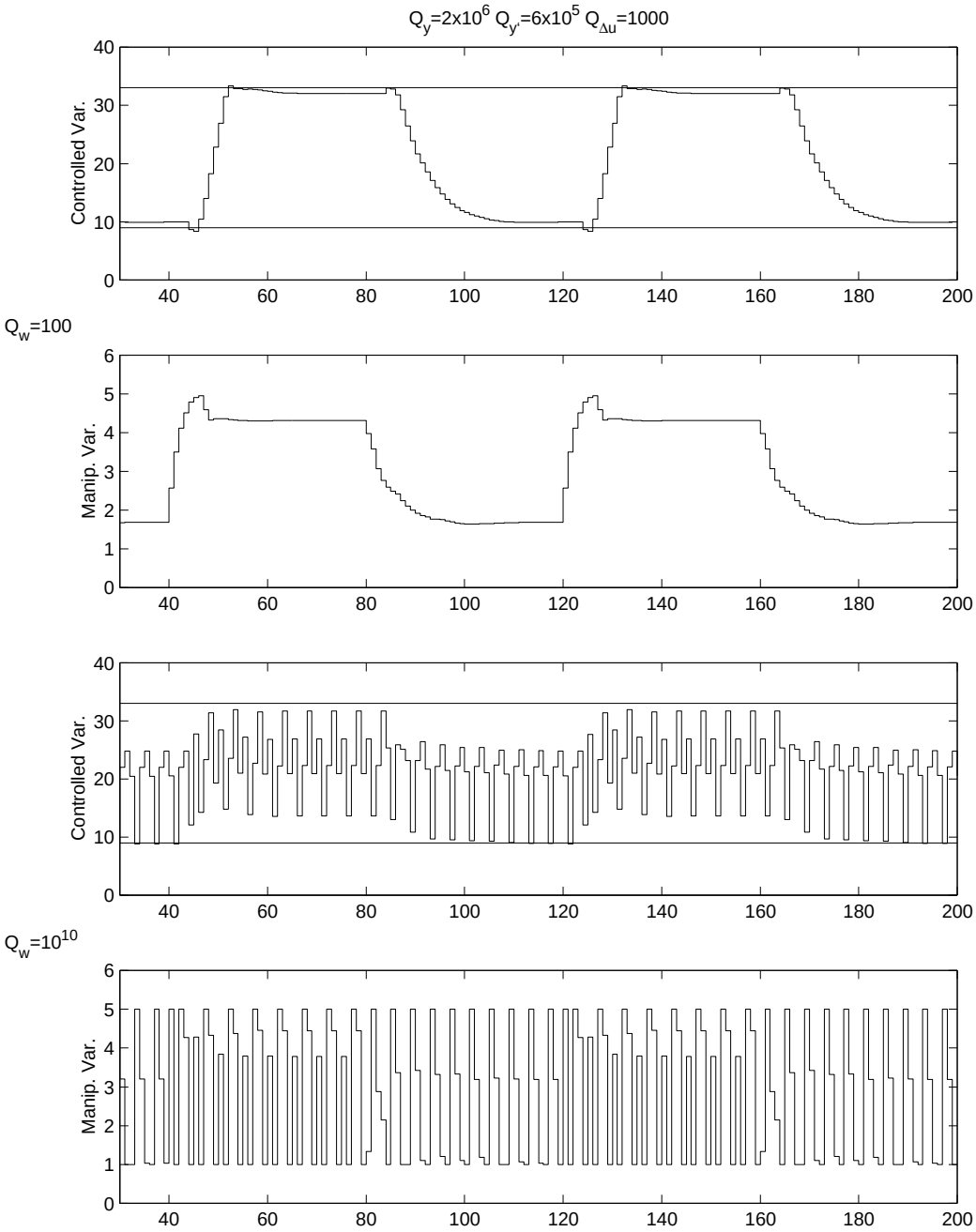

Figure 11: Effects of stability condition weighting Q_w on MoDMPC control performance in the presence of output constraints

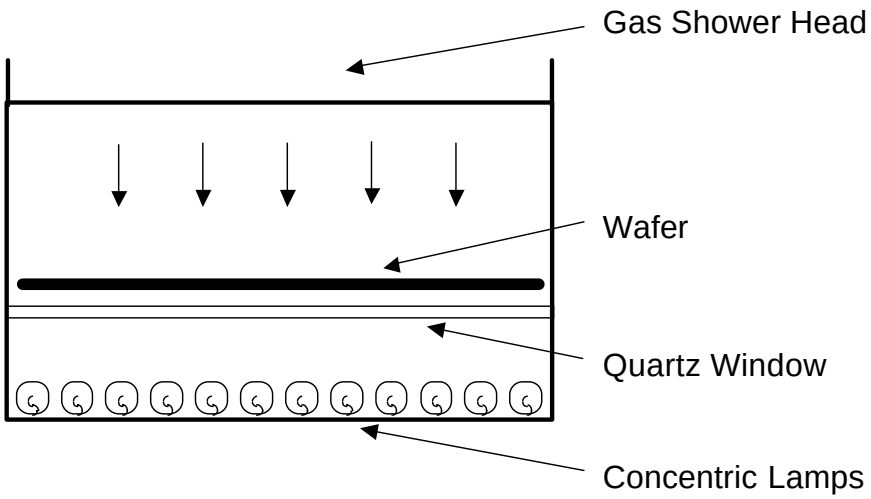

Figure 12: Schematic of 6×6 RTP wafer reactor

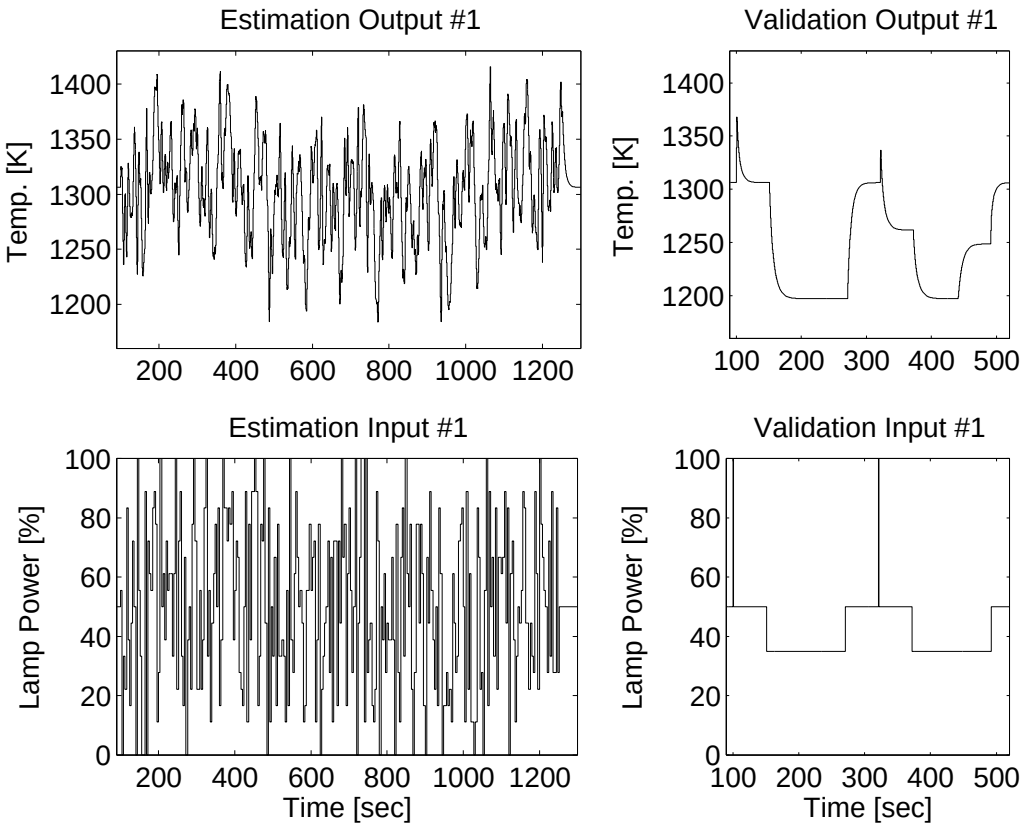

Figure 13: 17-level PRS input #1 and output #1

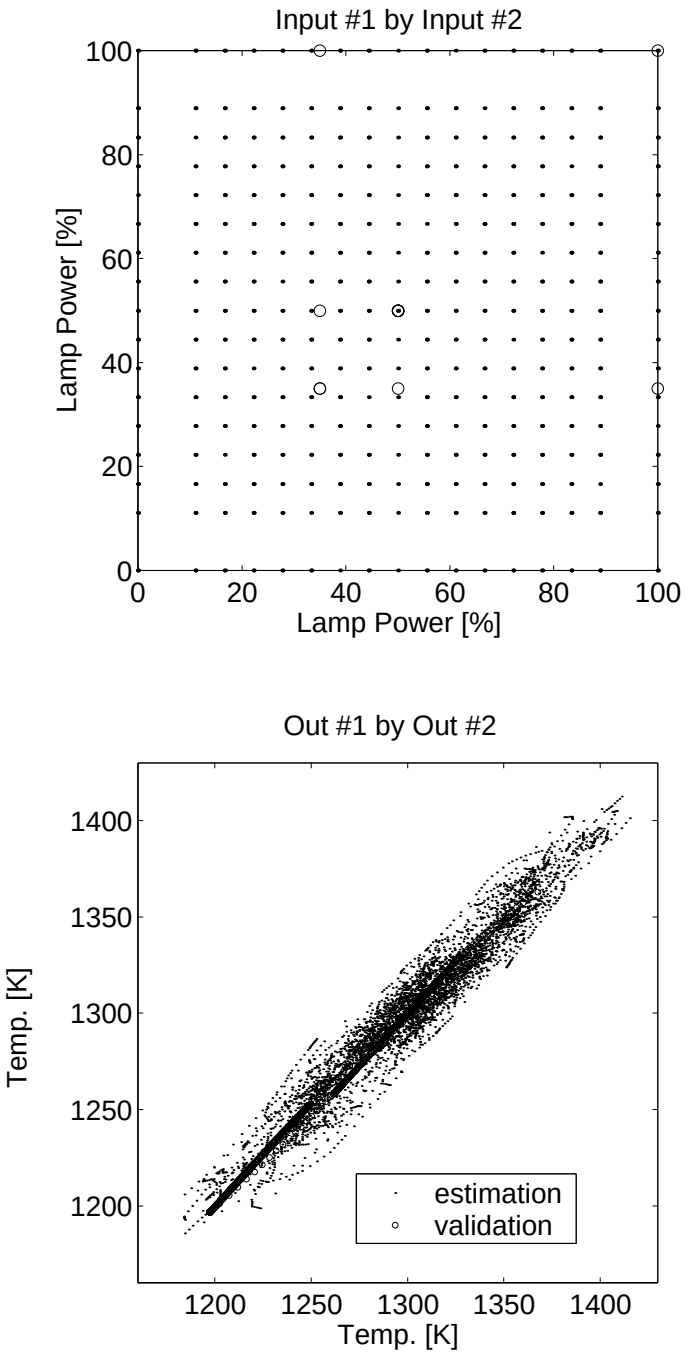

Figure 14: Input and output spaces for #1 and #2 combinations

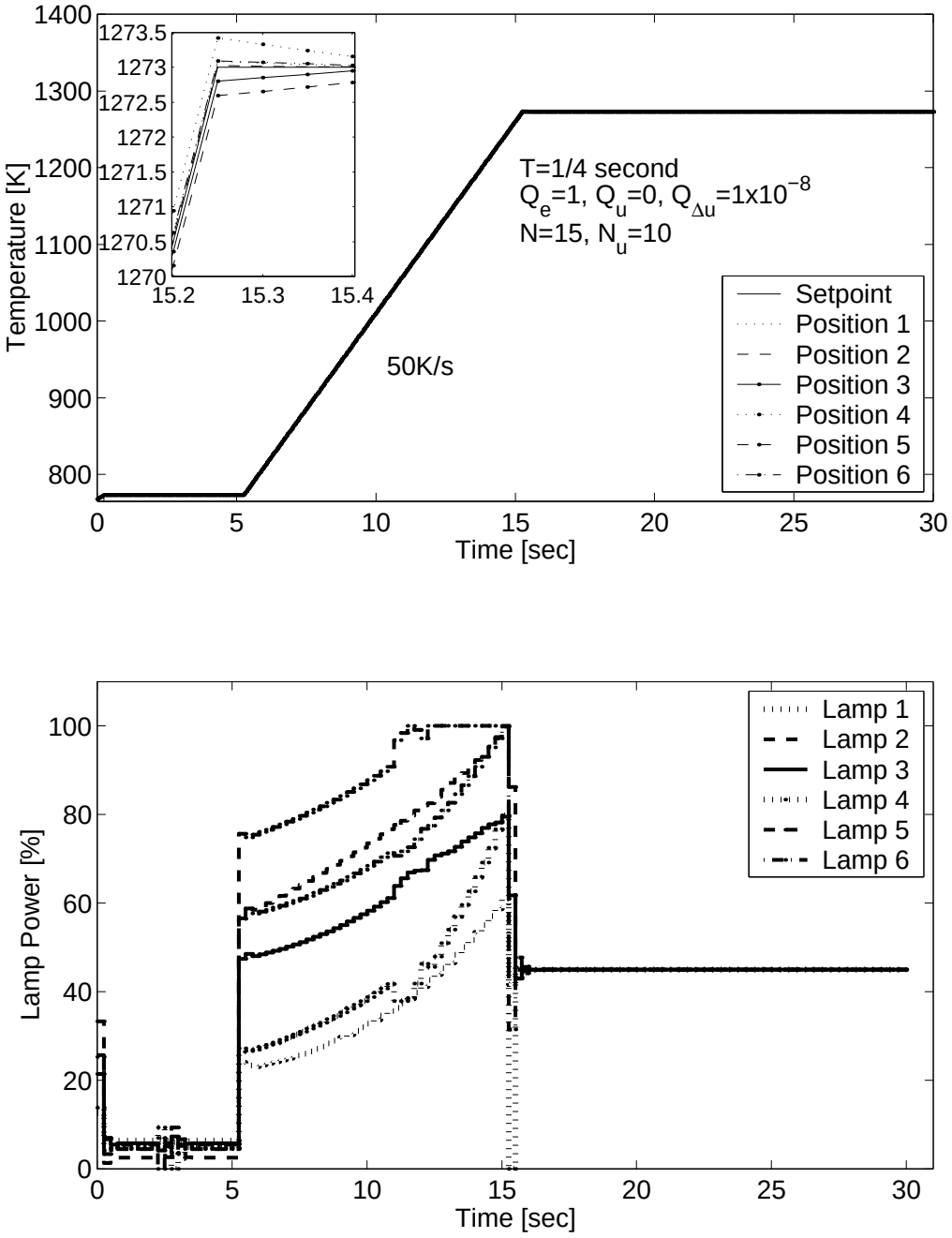

Figure 15: Control of 6x6 MIMO RTP reactor for 50 K/s ramp rate