S Y S T E M S
R E S E A R C H
C E N T E R

# Nonlinear Quadratic Dynamic Matrix Control with State Estimation

*by G. Gattu and E. Zafiriou*

TR 91-24r2

# Nonlinear Quadratic Dynamic Matrix Control with State Estimation

Gangadhar Gattu and Evanghelos Zafiriou*

Chemical Engineering Department
and Systems Research Center
University of Maryland
College Park, MD 20742

*Author to whom correspondence should be addressed. E-mail : zafiriou@src.umd.edu

# Abstract

Quadratic Dynamic Matrix Control (QDMC) with state estimation is presented for use with nonlinear process models. This formulation extends García's nonlinear version of QDMC to open-loop unstable nonlinear processes and allows for better disturbance rejection. Stability and better performance is observed when compared to the algorithm without state estimation in rejecting disturbances for processes operating at unstable steady state setpoints, as illustrated with two simple examples. The algorithm requires that only a Quadratic Program be solved on-line. The modest computational requirements make it attractive for industrial implementation. The effectiveness of the approach is demonstrated by its successful application to the temperature control of a semibatch polymerization reactor. A model and related control requirements for this problem were presented at the 1990 AIChE Annual Meeting in a session on "Industrial Challenge Problems in Process Control."

# 1 Introduction

Model Predictive Control (MPC) algorithms have been well received by the industry in the past few years because of their ability to handle input and output constraints and of their transparent tuning capabilities. Dynamic Matrix Control (DMC, Cutler and Ramaker, 1979) is one of the most popular algorithms among the model based control algorithms and it has been used extensively in the industry. The basic idea is to use a process model, in the form of step or impulse response coefficients, in parallel with the plant in order to predict future output values based on the past inputs (manipulated variables) and current measurements. An objective function is minimized on-line to compute the future manipulated variables. An extension of DMC to handle constraints explicitly as linear inequalities was introduced by García and Morshedi (1986) and denoted as Quadratic Dynamic Matrix Control (QDMC). Application of QDMC for processes which can be assumed linear has been mostly successful in the industrial environment for multivariable linear process models with input and output constraints (García and Prett, 1986).

Although linear MPC algorithms are successful in controlling processes which are linear and mildly nonlinear, performance degradation and instability often occur in the presence of strong nonlinearities. The advent of high speed computers provided enough motivation to propose algorithms that utilize nonlinear process models in the on-line optimization. As a result, a significant number of new control algorithms have been proposed based on nonlinear programming techniques (Brengel and Seider, 1989; Li and Biegler, 1989; Patwardhan et al., 1988). Peterson et al. (1990) update the linear model by computing a time varying disturbance vector which accounts for the nonlinearities. In their approach a Quadratic Program is solved repeatedly at each sampling point until the linear model approximates the output of the nonlinear model. Recent progress in nonlinear control is reviewed by Bequette (1990). However, the excessive computational requirements of such methods remain a serious obstacle to industrial implementation in spite of the advances made in developing efficient algorithms. Also, not much progress has been made in understanding their stability and performance properties.

Prior to the development of algorithms based on nonlinear programming techniques, García (1984) proposed an extension of QDMC to nonlinear processes. In his approach, the future manipulated variables are predicted at every sampling time. A nonlinear model is used to compute the effect of past manipulated variables and the future estimated disturbances on the predicted output. A linear model is obtained by linearization at each sampling time and used to compute the effect of future manipulated variables on the predicted output. Then, a Quadratic Program (QP) is solved as in the case of linear QDMC to compute the future manipulated variables. The requirement of solving only one QP on-line makes this algorithm an attractive option for industrial implementation. The major disadvantage of this algorithm is that it may not perform well in controlling integrating processes and may lead to instabilities when applied to open-loop unstable processes.

The main objective of this paper is to present an algorithm which is computationally feasible for industrial implementation and at the same time capable of handling open-loop unstable and integrating processes. This is accomplished by incorporating state estimation into

García's (1984) nonlinear version of QDMC. The steady state Kalman filter gain is computed at each sampling time by using a linearized model and a tuning parameter which represents the ratio between the uncertainty in the state and the noise in the measurement. Model states are compensated for unmodelled dynamics at each sampling time based on the measured variables and the steady state filter gain. The effectiveness of the proposed algorithm is demonstrated by its application to a marginally unstable system and to a completely unstable system. We also establish the connection between Ricker's (1990) state space formulation for linear QDMC and the proposed algorithm, when a linear model is used. Finally, the applicability of the algorithm on an industrial type problem is demonstrated by its successful application to the temperature control of a semibatch polymerization reactor. This process control problem was defined by Chylla and Haase (1990) in a session on "Industrial Challenge Problems in Process Control" at the 1990 Annual AIChE Meeting.

## 2  Linear and Nonlinear QDMC

The proposed algorithm is an extension of García's (1984) nonlinear version of QDMC (abbreviated to NLQDMC from here onwards). We give here a brief presentation of the QDMC concept and its extension to nonlinear processes. For a detailed description the reader is referred to García and Morshedi (1986) and García (1984).

In both QDMC and NLQDMC the computation at each sampling time involves the prediction of values of outputs in the future and the minimization of future squared output deviations from their setpoints to obtain the future manipulated variables. The output prediction is computed from the effect of the past and the future manipulated variables as well as the disturbance prediction into the future, which accounts for unmodelled effects and external process disturbances. Linear QDMC utilizes a step or impulse response model of the process and NLQDMC utilizes the model of the process represented by nonlinear ordinary differential equations. The difference between QDMC and NLQDMC lies only in the way the output values are predicted. Below we summarize how each contribution to the projected output is computed in NLQDMC. However, the approximations made in extending the QDMC to nonlinear processes such as superposition of past and future effects and constant step response coefficients for each of the future moves make the solution suboptimal. These approximations are necessary in order for the on-line optimization to be a single QP at each sampling point.

*Effect of past manipulated variables on the predicted outputs*

The model is described in the form of nonlinear differential equations. The contribution of the effect of past manipulated variables on the predicted outputs is defined as the value of the outputs if there are no input (manipulated variable) changes in the future. This is obtained by integrating the model differential equations from the current state over the prediction horizon with constant inputs.

*Effect of future manipulated variables on the predicted outputs*

The contribution of the future manipulated variables to the predicted output is represented

with the use of a step response model. A linear model obtained by linearization of the nonlinear model at each sampling time is used to compute the step response coefficients.

*Effect of future disturbances on the predicted outputs*

The unmodelled effects at the current sampling time are computed as the difference between the plant measurements and the model outputs. In the absence of any information on unknown disturbances in the future, it is assumed that the future predicted values of disturbances are equal to the current values.

# 3  Algorithm Formulation with State Estimation

Navratil *et al.* (1989) and Li *et al.* (1989) have proposed the use of state estimation with linear Model Predictive Controllers. Lee *et al.* (1990) generalized linear MPC to integrating systems and systems with measurement noise. Ricker (1990) formulated the objective function using discrete, linear, time invariant, state space models to take advantage of state estimation theory. In this paper we combine Ricker's (1990) approach for linear models with NLQDMC.

For the general case of Multi-Input Multi-Output (MIMO) systems, consider process and measurement models of the form

$$\dot{x} = f(x, u) + w \tag{1}$$

$$y = h(x) + v \tag{2}$$

where $x$ is the state vector, $y$ is the output vector, $u$ is the vector of manipulated variables and $w \sim (0, Q)$ and $v \sim (0, R)$ are white noise processes uncorrelated with each other. $Q$ and $R$ are covariance matrices associated with process and measurement noise respectively. In chemical processes the dynamic models are often poor and it is very difficult to characterize the disturbances. In the absence of reliable information on disturbance models, it is desirable to keep the structure of disturbance/noise as simple as possible. It is assumed that $Q \approx \sigma_w^2 I$ and $R \approx \sigma_v^2 I$, where $\sigma_w^2$ and $\sigma_v^2$ are scalar variances. Define $\sigma = \sigma_w/\sigma_v$ and let $\sigma_v^2 = 1$. The ratio of $\sigma_w^2$ to $\sigma_v^2$ determines the value of the Kalman filter gain. Intuitively, $\sigma$ is the ratio between statistical measures of the uncertainty in the state and the uncertainty in a measurement. Therefore, $\sigma$ can be used as a tuning parameter to give stability and robustness in the presence of model-plant mismatch, external disturbances and measurement noise. A large value of $\sigma$ implies that the model uncertainty dominates over measurement noise. Ricker (1990) demonstrated the effectiveness of a similar kind of tuning parameter for linear models.

**Algorithm**

*Known at sampling instant $k$ :*

$y(k)$ the plant measurement, $\hat{x}(k|k-1)$ the estimate of state vector at $k$ based on information at $k-1$, and $u(k-1)$ the manipulated variable.

*Effect of future manipulated variables*

**Step 1:** Linearize the nonlinear model $\dot{x} = f(x, u)$ at $\hat{x}(k|k-1)$ and $u(k-1)$ to obtain

$$\begin{aligned}
\dot{\hat{x}} &= A_k \hat{x} + B_k u \\
y &= C_k \hat{x}
\end{aligned} \qquad (3)$$

where

$$\begin{aligned}
A_k &= (\frac{\partial f}{\partial x})|_{x=\hat{x}(k|k-1), u=u(k-1)} \\
B_k &= (\frac{\partial f}{\partial u})|_{x=\hat{x}(k|k-1), u=u(k-1)} \\
C_k &= (\frac{\partial h}{\partial x})|_{x=\hat{x}(k|k-1)}
\end{aligned}$$

The computation of the derivatives can of course be made analytically. This however, is impractical, except for small systems. In general, numerical differentiation should be used. In the examples in this paper, the 5-point central difference formula was used, without any problems.

**Step 2:** Discretize (3) to obtain

$$\begin{aligned}
\hat{x}_{j+1} &= \Phi_k \hat{x}_j + \Gamma_k u_j \\
y_j &= C_k \hat{x}_j
\end{aligned} \qquad (4)$$

where $\Phi_k$ and $\Gamma_k$ are discrete state space matrices (e.g., Åström and Wittenmark, 1984), obtained from $A_k, B_k$ and the sampling time.

**Step 3:** Compute the step response coefficients $S_{i,k}$ ($i = 1, 2, \ldots, P$) where $P$ is the prediction horizon. Step response coefficients are used only to represent the effect of future manipulated variables. Therefore, only $P$ step response coefficients are required. $S_{i,k}$ can be obtained from

$$S_{i,k} = \sum_{j=1}^{i} C_k \Phi_k^{j-1} \Gamma_k \quad (i = 1, 2, \ldots, P) \qquad (5)$$

$S_{i,k}$ is of dimension $n_o \times n_i$, where $n_o$ is the number of outputs and $n_i$ is the number of inputs. Step response coefficients can also be obtained by numerical integration of the linearized model, since they are defined as the values of output for a constant unit step change in the input. Therefore, they can be obtained by successively integrating (1) over $P$ sampling intervals with $u = 1.0$ and $x(t_k) = 0.0$ where $t_k$ is the time at any sampling point $k$. The software package DDASSL (Petzold, 1983) was used for integration.

The contribution of the future manipulated variables to the predicted output values at $k+l$ is represented as $\sum_{i=1}^{l} S_{i,k} \Delta u(k+l-i)$ ($l = 1, 2, \ldots, P$), where $\Delta u$ is the change in manipulated variables, defined as $\Delta u(k) \stackrel{\Delta}{=} u(k) - u(k-1)$.

*Computation of filter gain*

**Step 4:** Compute the steady state Kalman gain using the recursive relation (Åström and Wittenmark, 1984):

$$\begin{aligned}
P_{(j+1)k} &= \Phi_k [P_{jk} - P_{jk} C_k^T (C_k P_{jk} C_k^T + R)^{-1} C_k P_{jk}] \Phi_k^T + Q \qquad (6) \\
K_k &= \Phi_k P_{\infty k} C_k^T [C_k P_{\infty k} C_k^T + R]^{-1} \qquad (7)
\end{aligned}$$

where $P_{jk}$ is the state covariance at iteration $j$ for the model obtained by linearization at sampling point $k$. $P_{\infty k}$ is the steady state value of state covariance for that model.

*Effect of past manipulated variables*

**Step 5:** The effect of past inputs on future output predictions, $y^*(k+1), y^*(k+2), \ldots, y^*(k+P)$ is computed as follows. Here the superscript '$*$' indicates that input values in the future are kept constant and equal to $u(k-1)$.

- Set $\hat{x}^*(k|k-1) = \hat{x}(k|k-1)$.

- Define $d(k|k)$ as $y(k) - h(\hat{x}(k|k-1))$.

- In the absence of measurement information in the future, it is assumed that $d(k+i|k) = d(k|k)$ for $i = 1, 2, \ldots, P$.

- For $i = 1, 2, \ldots, P$, successively integrate $\dot{x} = f(x, u)$ over one sampling time from $\hat{x}^*(k+i-1|k-1)$, with $u(k+i-1) = u(k-1)$ and then add $K_k d(k|k)$ to obtain $\hat{x}^*(k+i|k-1)$. Addition of $K_k d$ provides correction to the state. We can then write

$$y^*(k+i) = h(\hat{x}^*(k+i|k-1)) \quad (i = 1, 2, \ldots, P) \tag{8}$$

*Output Prediction*

**Step 6:** The predicted output is computed as the sum of the effect of past and future manipulated variables and the future predicted disturbances.

$$\hat{y}(k+l) = y^*(k+l) + \sum_{i=1}^{l} S_{i,k} \Delta u(k+l-i) + d(k|k) \quad (l = 1, 2, \ldots, P) \tag{9}$$

Even though the model states are compensated for unmodelled effects, the future predicted disturbances are added to the effect of past and future manipulated variables to compute the future predicted outputs. This is necessary in order to eliminate steady state offset. Ricker (1990) adds the disturbance vector $d(k|k)$ to the output prediction indirectly, by adjusting the reference setpoint used in the objective function. The reader can find there a discussion why this is necessary for avoiding steady-state offset. In a different form of state space formulations (Lee *et al.* 1990; Ricker, 1991), where outputs are also treated as states, this addition is not required. No straightforward incorporation of that approach to the original NLQDMC algorithm seems possible though.

*Optimization*

$$\min_{\Delta u(k),\ldots,\Delta u(k+M-1)} \sum_{l=k+1}^{k+P} (\hat{y}(l) - r(l))^T \Gamma^2 (\hat{y}(l) - r(l)) + \Delta u(l-1)^T \Lambda^2 \Delta u(l-1) \tag{10}$$

where $P$ is the prediction horizon and $M$ is the number of future moves to be optimized. It is assumed that $u(k+M-1) = u(k+M) = \ldots = u(k+P-1)$. $\Gamma$ and $\Lambda$ are diagonal weight matrices.

7

The above optimization problem with constraints can be written as a standard Quadratic Programming problem, as shown in García and Morshedi (1986):

$$\min_{X} \Phi(X) = \frac{1}{2}X^T G X + g^T X \tag{11}$$

subject to:

$$D^T X \geq b \tag{12}$$

where

$$X = [\Delta u(k) \dots \Delta u(k + M - 1)]^T \tag{13}$$

and $D$ and $b$ depend on the constraints on manipulated variables, change in manipulated variables and outputs.

**Step 7:** The $M$ future manipulated variables are computed, but only the first move is implemented (García and Morshedi, 1986).

*Estimation of state*

**Step 8:** Integrate $\dot{x} = f(x, u)$ from $\hat{x}(k|k - 1)$ and $u(k)$ over one sampling time and add $K_k d$ to obtain $\hat{x}(k + 1|k)$.

The above steps are repeated at each sampling time. For $\sigma = 0.0$, $K_k$ is 0.0 and this algorithm is equivalent to García's (1984). Ricker (1990) used $Q = \sigma_w^2 \Gamma_k \Gamma_k^T$ in his formulation for linear models. For such $Q$ and for linear models, our algorithm is equivalent to Ricker's without reference model dynamics. Incorporation of reference model dynamics is straightforward.

We formulated the algorithm making an assumption that the process and measurement noises are white noise processes with zero mean. However, the algorithm is also applicable for step like disturbances. We have tested the proposed algorithm with step like disturbances in input and demonstrated in examples 1 and 2. For step like disturbances, the plant states are not tracked perfectly by the model states, by using the proposed estimator. However, the purpose of the estimator in this paper is to make the closed loop system stable. The idea is similar to the design of an observer for linear systems. The asymptotic disturbance rejection property results from the addition of disturbance vector to predicted values of the output. We expect similar behavior when the system is subjected to colored noise.

# 4  Examples

In this section, the performance of the proposed algorithm on a marginally unstable system and on a completely unstable system is demonstrated. It is shown that NLQDMC can stabilize the marginally unstable system but not the completely unstable system, whereas NLQDMC with state estimation can stabilize both systems with excellent performance.

**Example 1** (*Marginally unstable system*)

This example problem is adapted from the paper by Li and Biegler (1988), which describes a kinetic model of a catalytic reaction in a CSTR with multiequilibrium points at steady

state. For the reaction $A + B \rightarrow P$ the rate of decomposition of B is

$$r_B = \frac{k_1 C_B}{(1 + k_2 C_B)^{2.0}} \qquad (14)$$

The system is described by a dynamic model of the form:

$$\frac{dx_1}{dt} = u_1 + u_2 - 0.2x_1^{0.5} \qquad (15)$$

$$\frac{dx_2}{dt} = (C_{B1} - x_2)\frac{u_1}{x_1} + (C_{B2} - x_2)\frac{u_2}{x_1} - \frac{k_1 x_2}{(1 + k_2 x_2)^{2.0}} \qquad (16)$$

$$y_1 = x_1 \qquad (17)$$

$$y_2 = x_2 \qquad (18)$$

where $u_1$ is the inlet flow rate with condensed B, $u_2$ is the inlet flow rate with dilute B, $x_1$ is the liquid height in the tank and $x_2$ is the concentration of B in the reactor. The control problem is simulated with the values $k_1 = 1.0$ , $k_2 = 1.0$, $C_{B1} = 24.9$ and $C_{B2} = 0.1$.

The model has two stable and one unstable steady state points. At the unstable steady state the linear model has the eigenvalues at $-0.01$ and $0.01286$. The model is marginally unstable with respect to the concentration. Here, we choose the unstable steady state point as setpoint to demonstrate the control of open-loop unstable systems using the proposed algorithm.

Figs. 1 and 2 illustrate the response of the reactor for a setpoint change from an initial condition of $x_{10} = 40.00$ and $x_{20} = 0.1$ to the unstable steady state point with values at $x_1 = 100.00$ and $x_2 = 2.787$. Here, we assume there is no model-plant mismatch. For no model-plant mismatch and no external disturbances the algorithms with and without state estimation result in identical responses. The lower bounds on $u_1$ and $u_2$ are kept at zero and the upper bounds varied from 5 to $\infty$. A sampling time T of 1.0 min and the tuning parameter values $P = 5$ and $M = 5$ are used in the simulations. For the tuning parameter $\Lambda$, a value of 0.0 is used when the upper bounds on manipulated variables are at 10.0 and $\infty$, and a value of 0.5 when the upper bounds are at 5.0. The results are comparable with the results presented by Li and Biegler (1988). Brengel and Seider (1989) also compare the results of their algorithm with those of Li and Biegler (1988). They claim marked improvement in reducing both overshoot and settling time. However, the proper choice of tuning parameters can also reduce the overshoot and settling time. For comparison purposes, simulations are presented with the same tuning parameters and sampling time chosen by Brengel and Seider. A sampling time of 1.0 min and tuning parameter values $P = M = 1$, $\Lambda = 0.0$, and $\Gamma = diag\{1, 20\}$ were used in their simulations. The manipulated variables were bounded between 0.0 and 10.0. The simulations with NLQDMC are given in Figs. 3 and 4. From Fig. 4, it can be seen that a large value of weight on the second output leads to better performance. For a value of weight 1.0 on the second output the response is similar to that presented by Li and Biegler and for a value of 20.0 the performance is better than that observed by Brengel and Seider.

Figs. 5 and 6 demonstrate the disturbance rejection capability of the proposed algorithm. Here we assume that the plant is running at the unstable steady state. A sampling time of 1.0

9

min and the tuning parameter values $P = 5.0$, $M = 5.0$, $\Lambda = 0.0$ and $\sigma = 1.0$ are used in the simulations. The lower bounds on $u_1$ and $u_2$ are kept at zero and there are no upper bounds. For a step disturbance of 0.5 units in $u_1$, the proposed algorithm rejects the disturbance in just a few sampling units. It is interesting to analyze the response of outputs using NLQDMC without state estimation. At steady state, the linear model has a stable pole corresponding to $x_1$ and an unstable pole corresponding to $x_2$. If we design a one-step-ahead controller based on the unstable linear model, the system will generate an unbounded response for a bounded input signal. But, in the nonlinear case, against this intuition, a strange behavior is observed. Since, the pole corresponding to $y_1$ is stable, this output slowly reaches its setpoint (Fig. 5). On the other hand (Fig. 6), $y_2$ continues to grow. This happens because the model goes through the locally unstable region. But, surprisingly after few sampling instants, the response turns around. Here, one should recall that, at each sampling time the nonlinear model is linearized at the values of the model states. Initially at steady state the linearized model has an unstable eigenvalue. Once the input disturbance is introduced, there is a mismatch between model and plant states and the model states take different values than the plant states. Initially the response of the closed-loop system is unstable because the linearized model has an unstable eigenvalue. After a few sampling instants, due to the nature of the disturbance, the linearized models have all eigenvalues stable. Although this is not sufficient by itself to stabilize the system, in this case it leads to closed-loop stability. This is an example where, even though the system is open-loop unstable, it happens that the nature of the disturbance makes the closed-loop system stable. This phenomenon may or may not be observed in other cases. It does not happen in example 2.

**Example 2** *(Completely unstable system)*

This example is adapted from the paper by Bruns and Bailey (1975). The system is described by a dynamic model of the form

$$\frac{dx}{dt} = u - x - \beta \frac{1.0}{1.0 + \frac{K_1}{x} + \frac{x}{K_2}} \tag{19}$$

$$y = x \tag{20}$$

where $y$ is the reactant concentration, $t$ is the dimensionless time, $u$ is the feed reactant concentration, $K_1$ and $K_2$ are kinetic constants and $\beta$ is a constant. The model for the single enzyme-catalyzed reaction with substrate inhibited kinetics (O'Neill et al., 1971) as well as the model for the ethylene hydrogenation in an isothermal CSTR (Matsuura and Kato, 1967) are of the above form. The reactant concentration in the reactor is controlled using the feed reactant concentration as the manipulated variable. The simulations are made using the values $\beta = 2.0$, $K_1 = 0.01$ and $K_2 = 0.1$.

The model has two stable and one unstable steady state points. The unstable steady state is at $x = 0.1250$ and $u = 0.9834$. The linear model at this steady state has the eigenvalue at 2.4482. The control objective is to keep the reactant concentration at its setpoint in the presence of input disturbances.

Figs. 7, 8 and 9 give the responses for a step input disturbance of 0.05 in the feed reactant concentration. In all simulations a lower bound of 0.0 is imposed on the input and there is

10

no upper bound. A sampling time of 0.1 is used. State estimation NLQDMC can reject the disturbance in a few sampling times as demonstrated in Fig. 7. Better performance is observed for a larger value of $\sigma$. If there are no unmeasured disturbances and the measurement noise is negligible, large values of sigma will lead to better performance. However, in the presence of measurement noise and unmeasured disturbances a large value of $\sigma$ may lead to instability. On the other hand, as shown in Figs. 8 and 9 NLQDMC without state estimation cannot stabilize the process and the output oscillates. The reason for this oscillation is the following. A linearized model at any state $x_{k0}$ and input $u_{k0}$ takes the form

$$
\begin{aligned}
\frac{dx}{dt} &= ax + u \\
a &= -1.0 + c \\
c &= \frac{2.0(10.0 - \frac{0.01}{x_{k0}^2})}{(1.0 + \frac{0.01}{x_{k0}} + 10.0x_{k0})^2}
\end{aligned}
\tag{21}
$$

The sign of the eigenvalue changes when $x_{k0}^2 < 0.001$ or $c < 1.0$ (which occurs when $x_{k0} > 0.342$). The model states oscillate between the stable and unstable regions and cannot be stabilized without the Kalman filter.

# 5 Temperature Control of a Semibatch Emulsion Polymerization Process

In this section we demonstrate the effectiveness of the NLQDMC algorithm for industrial implementation by its application to a temperature control of a semibatch polymerization reactor. The process description and control problem specifications are summarized from the paper by Chylla and Haase (1990).

## Process description

A stirred tank reactor is used to make specialty emulsion polymers in semibatch mode. The reactor temperature is maintained at its setpoint by adjusting the temperature setpoint for the water recirculating through the reactor jacket. In the current industrial implementation this is accomplished by a PID master controller. The temperature of the water recirculating through the reactor jacket is controlled to setpoint using a single input/dual output PID slave controller. When the controller output is between 0% and 50%, the controller is in cooling mode with the dump valve open 100% and 0% respectively. When it is between 50% and 100%, the controller is in heating mode and the steam valve open from 0% to 100%.

## Process Model

Chylla and Haase (1990) give material and energy balances in the form of nonlinear differential equations. Also, they provide recirculation loop dynamics in the form of transfer functions obtained by step tests on slave controller. Since a differential equation model is used in NLQDMC, we have to derive equivalent linear differential equations that include dead time to represent the recirculation loop dynamics. We transform the transfer function

11

representation of the recirculation loop dynamics into a state space representation. This is illustrated below. Then, we augment the linear state space equation with the nonlinear differential equations of the material and energy balances. For the material and energy balance equations and parameters the reader is referred to Chylla and Haase (1990). The transfer functions representing the recirculation loop dynamics relate the % change in the span of the instrument $(100^\circ F - 300^\circ F)$ which measures the jacket temperature $T_j$ to a step change in the controller output (Chylla, 1991). Two different transfer functions are provided for the recirculation loop dynamics for the heating and cooling modes. For example, the transfer function for the heating mode is

$$G(s) = \frac{0.42e^{-0.1s}}{5.1s + 1} \tag{22}$$

This equation corresponds to

$$\frac{dx(t)}{dt} = -\frac{1.0}{5.1}x(t) + \frac{0.42}{5.1}u(t - 0.1) \tag{23}$$

$$T_j(t) = (300.0 - 100.0)\frac{x(t)}{100.0} + T_{j0} \tag{24}$$

where $T_{j0}$ is the initial value of the jacket temperature, $x$ is the % change in the span of the instrument which measures $T_j$, and $u$ is the deviation variable of the controller output with respect to its initial value $u_0$. The value $u_0 = 50.0$ is used in the simulations. For $t < 0.1$, $u$ is set equal to $u_0$. Similarly, a state space equation can be derived for the cooling mode.

**Control Objectives**

- The reactor temperature should be maintained at its setpoint throughout the batch. Deviations of less than $1^\circ F$ are considered satisfactory.

- During successive batches of the same product (the reactor is used to manufacture different products), a polymer film forms on the reactor walls, which gets thicker with each batch. The controller with the same values of tuning parameters must perform well over a series of five consecutive batches under both winter and summer conditions.

A minimum change in controller output of 1% (2% valve position) is necessary in order to overcome friction in the valve stem and hence to change the valve position.

Chylla and Haase (1990) have given recipes and data for products A and B. (A typographical error (Chylla, 1991) exists on page 16 of Chylla and Haase (1990). The data $UA_{loss} = 1074$ $Btu \ h^{-1} \ ^\circ F^{-1}$ should be read as $UA_{loss} = 10.74 \ Btu \ h^{-1} \ ^\circ F^{-1}$.) In this paper we only give simulations for the case of product A. The performance of the algorithm was equally satisfactory for product B. The recipe for product A is as follows.

(i) Place initial charge of solids and water into the reactor at ambient temperature.

(ii) Raise the temperature of the initial charge to the reaction temperature setpoint.

(iii) Feed pure monomer into the reactor at 1.0 lb/min for 70 minutes.

(iv) After the feed addition period is complete, hold at reaction temperature for 60 minutes.

## Simulations with NLQDMC

The process never operates at locally open-loop unstable regions, so the Kalman filter is not needed for stability, although it can be used. The manipulated variable is the controller output which sends signal to either the steam or the dump valve, depending on whether the controller is in heating mode or in cooling mode. The process output is the reactor temperature, which is the measurement fed to the NLQDMC algorithm together with its desired (setpoint) value. We assume that at $k = 0$, the temperature of the initial charge has been raised to its setpoint ($180^0 F$). There is no reaction until the monomer addition begins. In the absence of reaction, the jacket temperature has to be maintained at a higher value than that of the reactor because of the heat losses to surroundings. From the energy balance, it was found that the jacket temperature should be maintained at a value somewhere between $180.6^0 F$ and $181.5^0 F$ depending on the degree of fouling and ambient conditions. The controller will be in heating mode until the value of heat generation due to reaction exceeds that of heat losses to surroundings. Since the heat generation due to reaction is large when compared to heat losses to surroundings, the controller switches to cooling mode almost immediately once the monomer addition begins. Therefore for simplicity in the simulations, it is assumed that the controller output is at 50% when the monomer addition begins (at time 0.0 in the simulations). A few cases are simulated to demonstrate that the controller works over a series of consecutive batches in the presence of fouling and under both winter and summer ambient conditions. A lower bound of 0.0 is imposed on the manipulated variable in all simulations. If the change in controller output is less than 1%, then the previous control move is implemented.

The material and energy balance equations augmented with (23) form a state space model with three states. Because the time delay appears in a linear manner, one can linearize the nonlinear differential equations first, ignoring the delay, and then simply include it at the input of the linearized model. This is then integrated numerically with DDASSL to produce the step response coefficients.

The tuning parameter values $P = 10, M = 5$ and $\Lambda = 0.1$ are used in the simulations. In all simulations a value of 1.0 is used for the impurity factor in the rate equation. The performance is equally good for any random value of impurity factor between 0.8 and 1.2 (as specified in Chylla and Haase, 1990). Figs. 10 to 13 demonstrate the temperature control under winter and summer ambient conditions. Simulations are shown with uncertainty in the fouling factor ($h_f$) as prescribed in Chylla and Haase (1990). In all cases the reactor temperature is always maintained at its setpoint with $\pm 0.2^0 F$ error, which is well within the $1.0^0 F$ maximum deviation allowed. One should note that although the reactor temperature remains very close to the setpoint, the process characteristics change quite substantially when the control system switches from cooling to heating mode. The gain of the linearized model almost doubles when that happens.

13

# 6 Conclusions

A nonlinear QDMC algorithm with state estimation was presented in this paper. For a special choice of a tuning parameter the proposed algorithm reduces to García's (1984) algorithm. For a special choice of covariance matrix associated with process noise and, for linear models, the algorithm is essentially equivalent to Ricker's (1990) linear state space formulation without reference model dynamics. A tuning parameter similar to that used by Ricker for linear models is effective for nonlinear models as well. The proposed algorithm stabilizes open-loop unstable plants and provides better disturbance rejection when compared to García's algorithm. For no modeling error and no external disturbances the proposed algorithm is equivalent to García's and the results are comparable with those presented by Li and Biegler (1988). Brengel and Seider (1989) claim better performance of their algorithm over Li and Biegler's. However, it was found that by appropriately choosing the tuning parameters both the results presented by Li and Biegler as well as Brengel and Seider can be reproduced or improved upon by using nonlinear QDMC with state estimation. The incorporation of a Kalman filter also results in better disturbance rejection for a marginally unstable system. In a completely unstable system, the proposed algorithm can reject the disturbance and drive the system to setpoint almost immediately, whereas García's algorithm cannot stabilize the process. One should note, however, that the assumptions on the trivial form of the covariance matrices result in a very simplified treatment of disturbance modeling. One can of course use more complex forms, but as the examples show this is usually not necessary. The effectiveness of the nonlinear QDMC algorithm is also demonstrated on the temperature control of a semi-batch emulsion polymerization process, which has been presented as an industrial challenge problem.

Other algorithms based on nonlinear programming techniques (Brengel and Seider, 1989; Patwardhan *et al.*, 1988; Schmid and Biegler, 1990) have also demonstrated the control of unstable plants. *The major advantage of the proposed algorithm compared to the nonlinear programming approaches is that only a single Quadratic Program is solved on-line at each sampling time, which makes the proposed algorithm an attractive option for industrial implementation.*

Conditions for robust stability and performance of the nonlinear control algorithms is still an open research issue. It is not clear, what are the merits and demerits of the various algorithms over the others for analysis purposes. Any breakthrough in this area will be a major accomplishment. The computational simplicity of the proposed algorithm may make the theoretical analysis simpler. Work in this direction is currently underway in our research group.

## Acknowledgements

14

Modified software from the package CONSYD, developed at Caltech (Dr. M. Morari's group) and the University of Wisconsin (Dr. W. H. Ray's group), was used in the simulations.

## Notation

| | |
|---|---|
| $a, c$ | constants |
| $b$ | inequality constraint equation vector |
| $A, B, C$ | continuous state space matrices |
| $D$ | inequality constraint equation matrix |
| $C_B$ | concentration of species B |
| $C_{B1}$ | species B in concentrated feed |
| $C_{B2}$ | species B in dilute feed |
| $d$ | disturbance |
| $g$ | the gradient vector |
| $G$ | hessian matrix |
| $G(s)$ | transfer function of recirculation loop |
| $h_f$ | fouling factor |
| $k$ | current sampling time index |
| $k + 1|k$ | estimate at k+1 based on information at k |
| $k_1, k_2, K_1, K_2$ | kinetic constants |
| $M$ | no. of future moves |
| $n_o, n_i$ | no. of process outputs and inputs |
| $P$ | prediction horizon |
| $P_{jk}$ | state covariance at iteration $j$ and sampling time $k$ |
| $P_{\infty k}$ | steady state state covariance at sampling time $k$ |
| $Q, R$ | covariance matrices |
| $r$ | set point |
| $r_B$ | rate of decomposition of B |
| $S_{i,k}$ | step response coefficient matrix |
| $t$ | time |
| $T$ | sampling time |
| $T_{amb}$ | ambient temperature |
| $T_j$ | jacket temperature |
| $u$ | manipulated variable |
| $UA_{loss}$ | rate of heat loss to surroundings |
| $w, v$ | white noise processes |
| $x$ | state |
| $X$ | vector of change in manipulated variables |
| $y$ | output |

*Greek letters*

| | |
|---|---|
| $\beta$ | constant |

| $\Gamma, \Lambda$ | diagonal weight matrices |
| $\Gamma_k, \Phi_k$ | discrete state space matrices |
| $\Delta$ | change in the associated variable |
| $\theta$ | time delay |
| $\sigma$ | ratio $\sigma_w / \sigma_v$ |
| $\sigma_w^2, \sigma_v^2$ | scalar variances |

*Subscripts*

| 0 | initial or nominal value |
| $k$ | current sampling time index |
| $k0$ | initial or nominal value at $k$ |

*Superscripts*

| $\char`^$ | estimated value |
| $\cdot$ | derivative |
| $T$ | transpose |
| $\star$ | represents the effect of past |

# References

[] Åström, K. J.; Wittenmark B. *Computer Controlled Systems: Theory and Design*, Prentice-Hall, Englewood Cliffs, NJ, 1984; chapters 3 and 11.

[] Bequette, B. W. "Nonlinear Control of Chemical Processes - A Review," submitted for publication in *Ind. and Eng. Chem. Res.*, 1990.

[] Brengel, D. D.; Seider W. D. "Multistep Nonlinear Predictive Controller," *Ind. and Eng. Chem. Res.*, **28**, pp. 1812-1822, 1989.

[] Bruns, D. D.; Bailey, J. E. "Process Operation Near an Unstable Steady State Using Nonlinear Feedback Control," *Chem. Eng. Sci.*, **30**, pp. 755-762, 1975.

[] Chylla, R. W. *personal communication*, 1991.

[] Chylla, R. W.; Haase, D. R. "Temperature Control of Semibatch Polymerization Reactors," *AIChE Annual Meeting*, Chicago, IL, 1990.

[] Cutler, C. R.; Ramaker, B. L. "Dynamic Matrix Control - A Computer Control Algorithm," *AIChE Annual Meeting*, Houston, TX, 1979; also in *Proceedings of Joint Automatic Control Conference*, San Francisco, CA, 1980.

[] García, C. E. "Quadratic/Dynamic Matrix Control of Nonlinear Processes: An Application to a Batch Reaction Process," *AIChE Annual Meeting*, San Francisco, CA, 1984.

[] García, C. E.; Morshedi, A. M. "Quadratic Programming Solution of Dynamic Matrix Control (QDMC)," *Chem. Eng. Comm.*, **46**, pp. 73-87, 1986.

[] García, C. E.; Prett, D. M. "Advances in Industrial Model Predictive Control," in *Chemical Process Control-CPC III*, Morari, M.; McAvoy, T. J., eds., CACHE and Elsevier, Amsterdam, 1986.

[] Lee, J. H.; Morari, M.; García, C. E. "State Space Interpretation of Model Predictive Control," submitted for publication in *Automatica*, 1990.

[] Li, W. C.; Biegler, L. T. "Process Control Strategies for Constrained Nonlinear Systems," *Ind. and Eng. Chem. Res.*, **27**, pp. 1421-33, 1988.

[] Li, W. C.; Biegler, L. T. "Multistep Newton-Type Controller Strategies for Constrained Nonlinear Processes," *Chem. Eng. Res. Des.*, **67**, pp. 562-577, 1989.

[] Li, S.; Lim, K. Y.; Fisher, D. G. "A State Space Formulation for Model Predictive Control," *AIChE J.*, **35**, pp. 241-249, 1989.

[] Matsuura, T.; Kato, M. "Concentration Stability of the Isothermal Reactor," *Chem. Eng. Sci.*, **22**, pp. 171-184, 1967.

[] Navratil, J. P.; Lim, K. Y.; D. G. Fisher, "Disturbance Feedback in Model Predictive Control Systems," in *Model Based Process Control-Proceedings of the 1988 IFAC Workshop*, pp. 63-68, McAvoy, T. J.; Arkun, Y.; Zafiriou, E., eds., Pergamon Press, Oxford, 1989.

[] O'Neill, S. P.; Lilly, M. D.; Rowe, P. N. "Multiple Steady States in Continuous Flow Stirred Tank Enzyme Reactors," *Chem. Eng. Sci.*, **26**, pp. 173-175, 1971.

[] Patwardhan, A. A.; Rawlings, J. B.; Edgar, T. F. "Model Predictive Control of Nonlinear Processes in the Presence of Constraints," *AIChE Annual Meeting*, Washington, DC, 1988.

[] Peterson, T.; Hernandez, E.; Arkun, Y.; Schork, F. J. "A Nonlinear DMC Algorithm and its Application to a Semi Batch Polymerization Reactor," submitted for publication in *Chem. Eng. Sci.*, 1990.

[] Petzold, L. R. "A Description of DASSL: A Differential-Algebraic System Solver," in *Scientific Computing*, pp. 65-68, R. S. Stepleman, ed., North-Holland, 1983.

[] Ricker, N. L. "Model Predictive Control With State Estimation," *Ind. and Eng. Chem. Res.*, **29**, pp. 374-382, 1990. Ricker, N. L. "Model Predictive Control: State of the Art," in *Chemical Process Control – CPC IV Proceedings*, South Padre Island, TX; Arkun, Y.; Ray, W. H. (eds.), CACHE, AIChE (publication No. P-67), New York, NY, 1991.

[] Schmid, C.; Biegler, L. T. "Application of Multistep Newton-Type Controllers to Fluid Catalytic Cracking," in *Proceedings of the American Control Conference*, pp. 581-586, San Diego, CA, 1990.
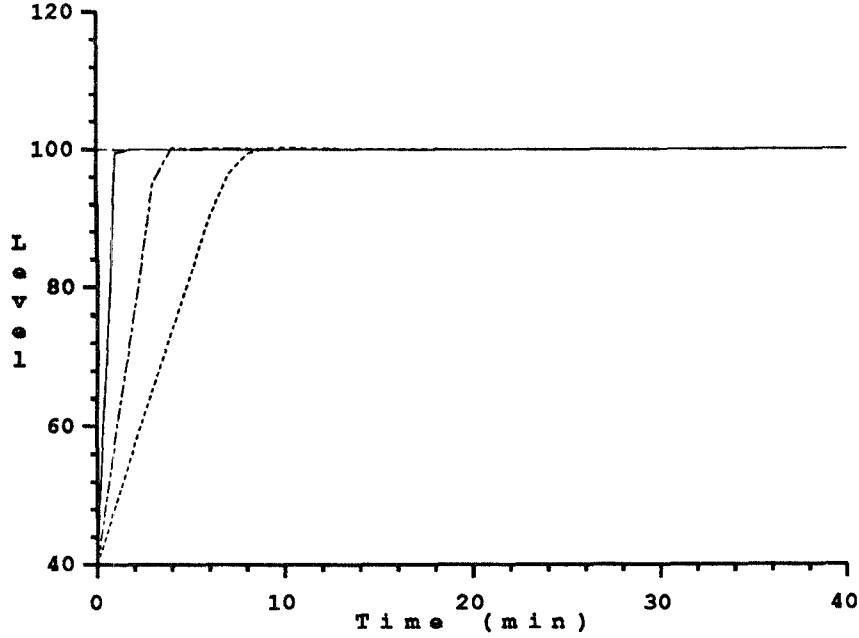
Figure 1: Level vs. time. Solid line –no upper bounds on $u_1$ and $u_2$; Alternate dots and dashes –upper bounds at 10.0; Dotted line –upper bounds at 5.0
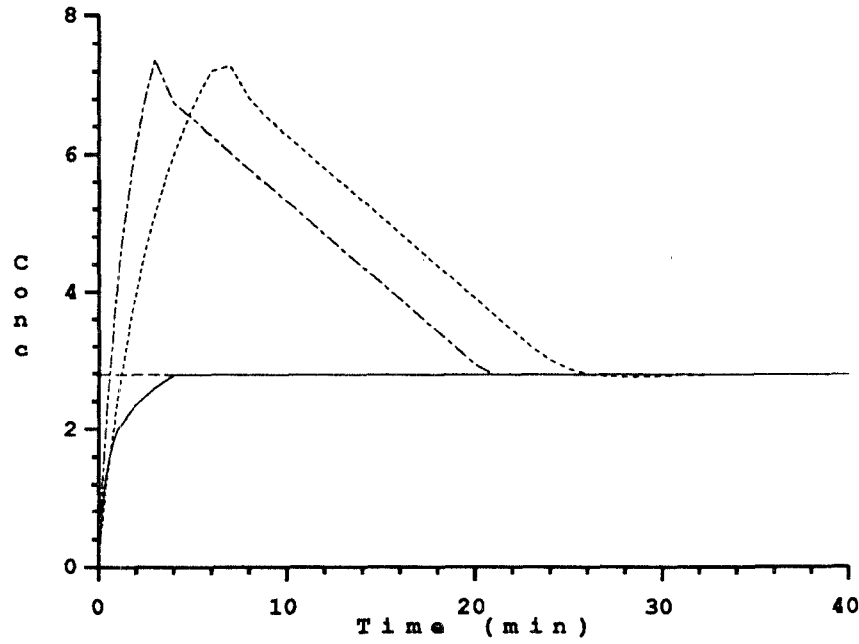


Figure 2: Concentration vs. time. Solid line –no upper bounds on $u_1$ and $u_2$; Alternate dots and dashes –upper bounds at 10.0; Dotted line –upper bounds at 5.0
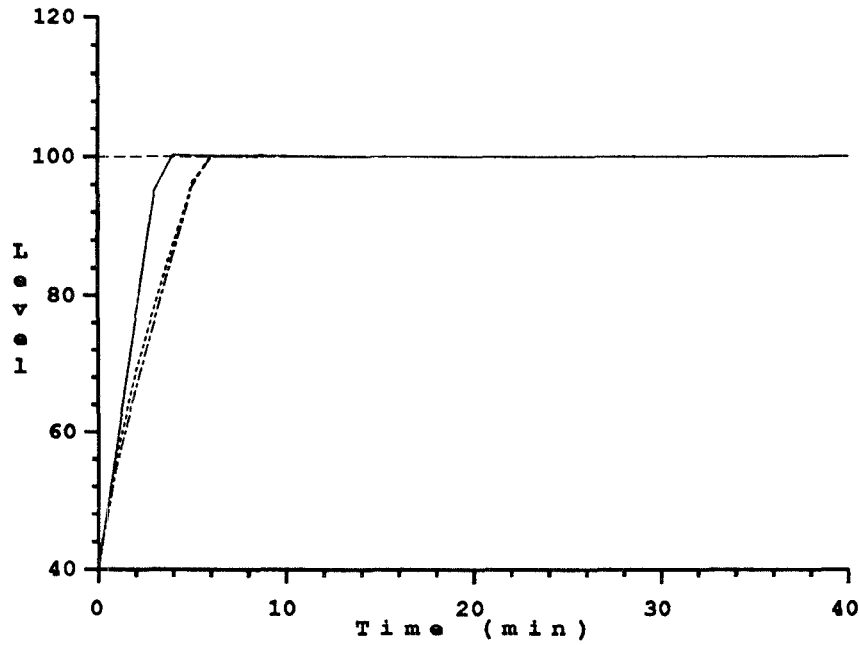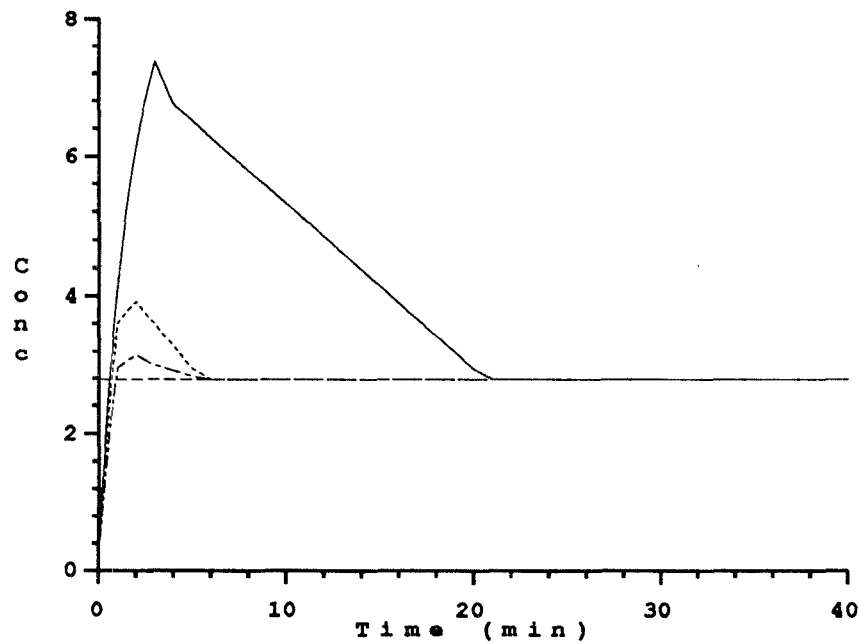
Figure 3: Level vs. time. Inputs are bounded between 0.0 and 10.0; Solid line —$\Gamma$ = diag$\{1,1\}$; Dotted line —$\Gamma$ = diag$\{1,10\}$; Alternate dots and dashes —$\Gamma$ = diag$\{1,20\}$



Figure 4: Concentration vs. time. Inputs are bounded between 0.0 and 10.0; Solid line —$\Gamma$ = diag$\{1,1\}$; Dotted line —$\Gamma$ = diag$\{1,10\}$; Alternate dots and dashes —$\Gamma$ = diag$\{1,20\}$
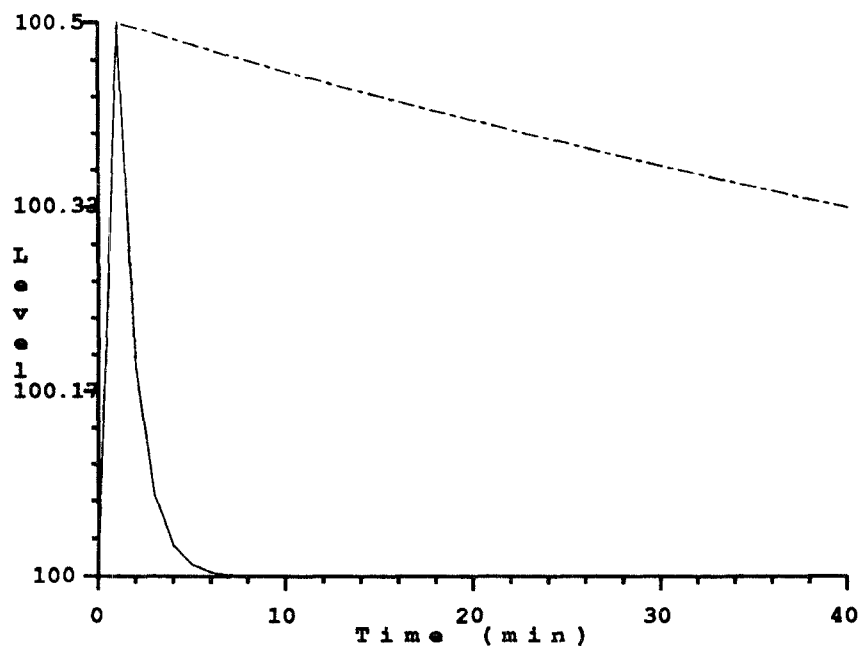
Figure 5: Level vs. time. A step disturbance of 0.5 units on $u_1$; Solid line –Nonlinear QDMC with state estimation; Alternate dots and dashes –Nonlinear QDMC without state estimation
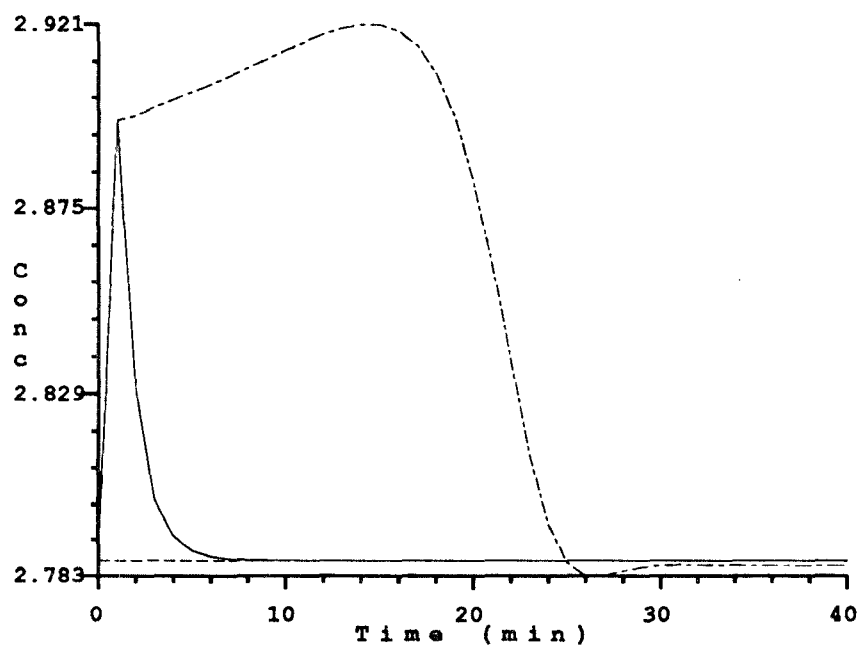


Figure 6: Concentration vs. time. A step disturbance of 0.5 units on $u_1$; Solid line –Nonlinear QDMC with state estimation; Alternate dots and dashes –Nonlinear QDMC without state estimation
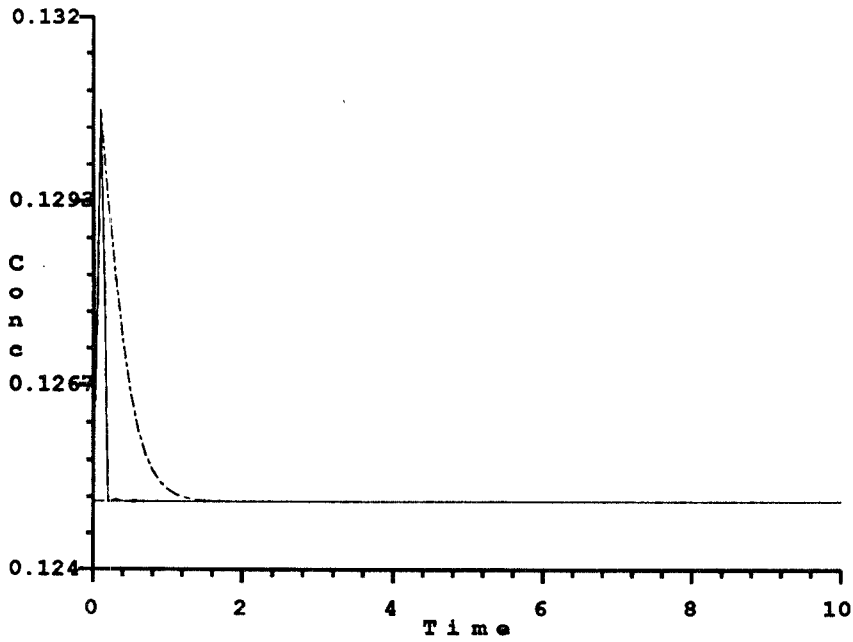
Figure 7: Concentration vs. time. A step disturbance of 0.05 in $u_1$; Nonlinear QDMC with state estimation; P = 5, M = 5 and $\Lambda$ = 0.0; Solid line $\sigma \to \infty$; Alternate dots and dashes $\sigma = 0.1$
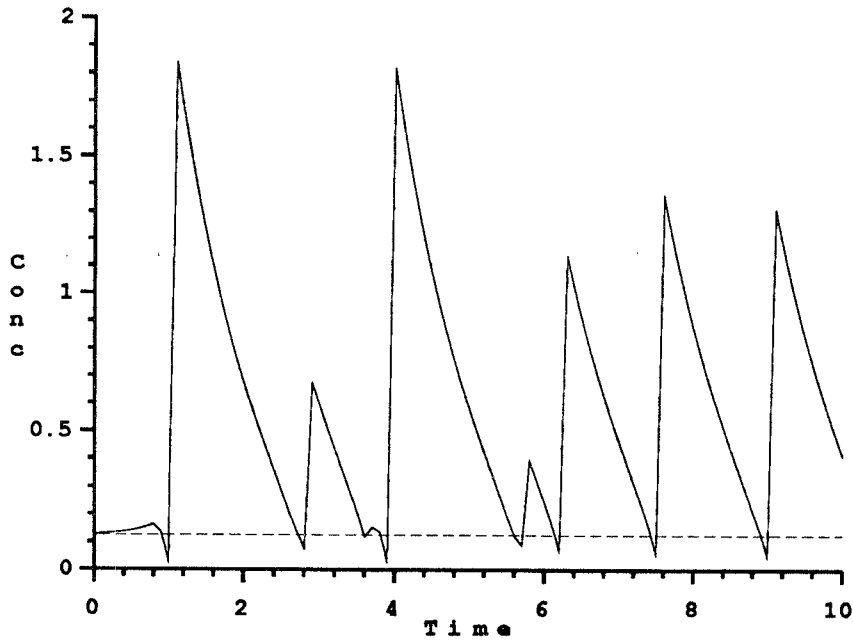


Figure 8: Concentration vs. time. A step disturbance of 0.05 in $u_1$; Nonlinear QDMC without state estimation; P = 5, M = 5 and $\Lambda$ = 0.0
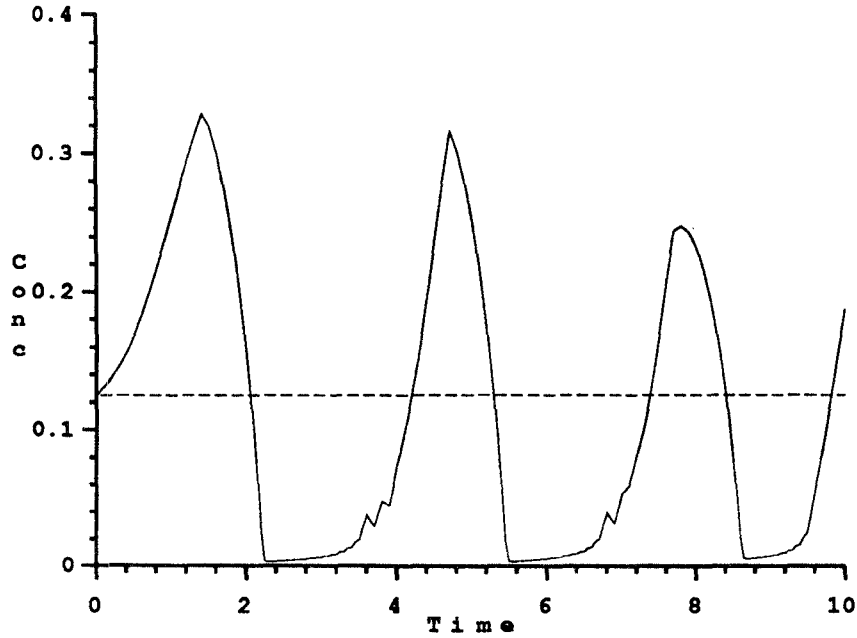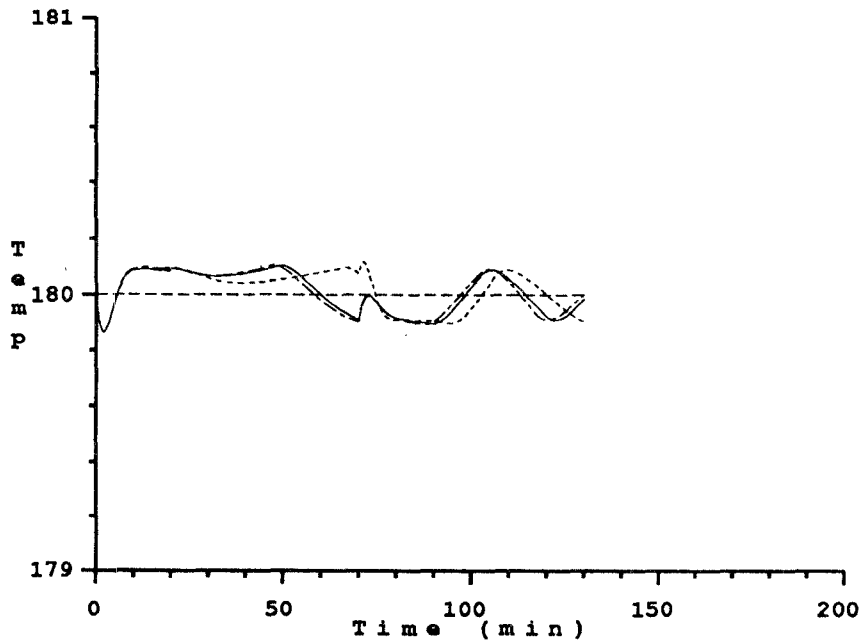
Figure 9: Concentration vs. time. A step disturbance of 0.05 in $u_1$; Nonlinear QDMC without state estimation; P = 10, M = 1 and $\Lambda$ = 0.5



Figure 10: Temperature vs. time; $T_{amb} = 45^0 F$. Dotted line $h_f = 0.000$; Solid line $h_f = 0.002$; Alternate dots and dashes $h_f = 0.004$.
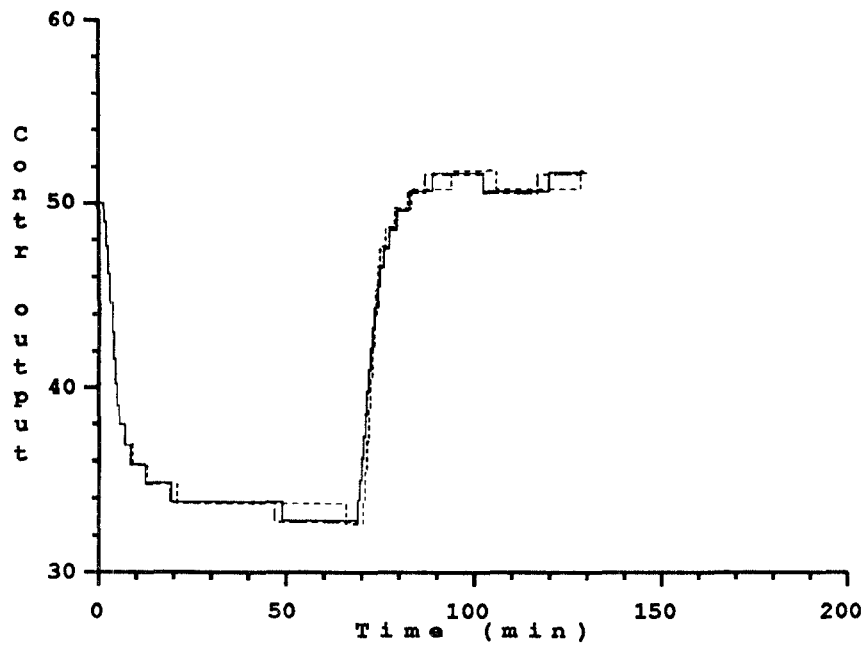
23

Figure 11: Controller output vs. time; $T_{amb} = 45^\circ F$. Dotted line $h_f = 0.000$; Solid line $h_f = 0.002$; Alternate dots and dashes $h_f = 0.004$.
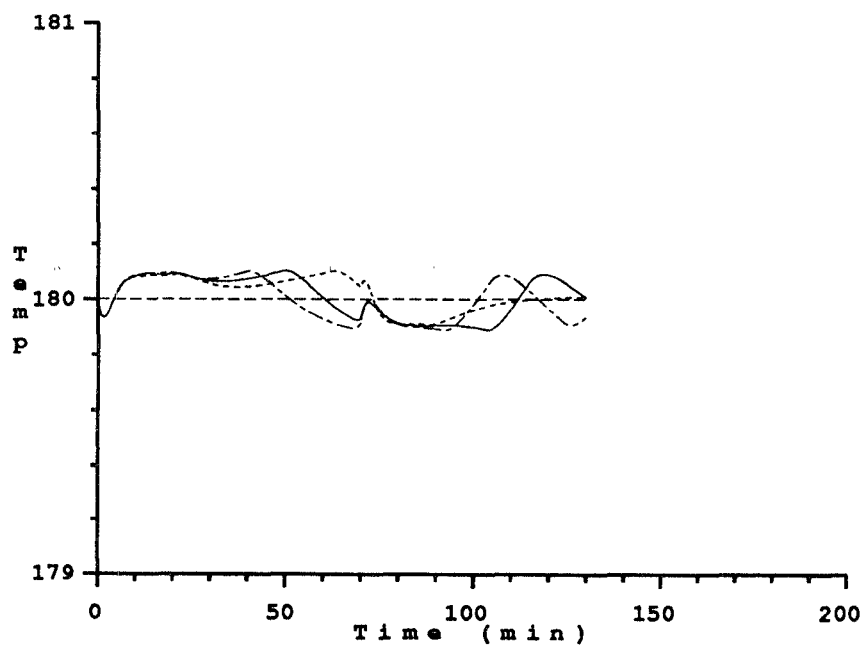


Figure 12: Temperature vs. time; $T_{amb} = 90^\circ F$. Dotted line $h_f = 0.000$; Solid line $h_f = 0.002$; Alternate dots and dashes $h_f = 0.004$.
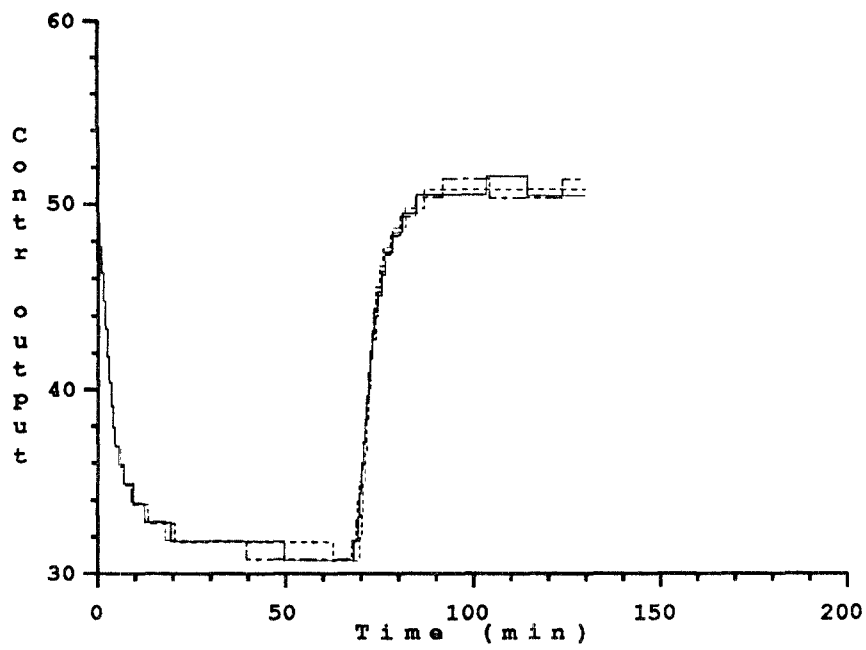
Figure 13: Controller output vs. time; $T_{amb} = 90^0 F$. Dotted line $h_f = 0.000$; Solid line $h_f = 0.002$; Alternate dots and dashes $h_f = 0.004$.