

TECHNICAL RESEARCH REPORT

State Estimation Nonlinear QDMC with
Input-Output Models

by G. Gattu and E. Zafiriou

T.R. 94-25



*Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry*

State Estimation Nonlinear QDMC with Input-Output Models¹

Gangadhar Gattu and Evangelhos Zafiriou²

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

Abstract

A State Estimation NLQDMC algorithm is presented for use with nonlinear input-output models. The proposed algorithm extends the state estimation NLQDMC [5] to nonlinear models identified based on input-output information. The algorithm preserves the computational advantages of [5] when compared to the other algorithms based on nonlinear programming techniques. The illustrating example demonstrates the usage of tuning parameters and points out the benefits and shortcomings of the algorithm.

1. Introduction

A significant number of Model Predictive Control (MPC) algorithms that utilize nonlinear process models in the on-line optimization have appeared in the literature. In all these algorithms an objective function is minimized to compute the future manipulated variables. The various algorithms based on nonlinear programming techniques (e.g., [11,12]) differ in the way that the ordinary differential equations are solved and in the optimization approach utilized. Garcia [3] proposed an extension of linear Quadratic Dynamic Matrix Control (QDMC) to nonlinear processes (abbreviated to NLQDMC from here onwards). Although a nonlinear model is used, only a single Quadratic Program (QP) is solved on-line. Gattu and Zafiriou [5] extended this formulation to open-loop unstable systems, by incorporating a Kalman filter. The requirement of solving only one QP on-line at each sampling time makes this algorithm an attractive option for industrial implementation. This extension of NLQDMC to open-loop unstable systems was *ad hoc* and did not address the problem of offset free tracking and disturbance rejection in a general state space setting. Independent white noise was added to the model states to handle unstable processes. The approach can stabilize the system but leads to an offset in the presence of persistent disturbances. To obtain offset free tracking Gattu and Zafiriou [5] added a constant disturbance to the predicted output as done in DMC-type algorithms. This addition is *ad hoc* and does not result from the filtering/prediction theory. This is also pointed out in [10].

There is a recent surge in the use of nonlinear models identified based on input-output information, for control purposes using the Model Predictive Control

schemes. Saint-Donat *et al.* [14] used neural network models in the on-line optimization. They solved the on-line optimization problem utilizing the nonlinear programming techniques. Hernandez and Arkun [8,9] used the neural network models and polynomial ARMA models in the on-line optimization. They used the extended DMC algorithm and the algorithm based on nonlinear programming techniques [7] for on-line control.

In this paper, we present an algorithm for use with nonlinear input-output models which addresses the offset free tracking problem and disturbance rejection problem in a general setting.

2. Algorithm

In this section, we present the state estimation NLQDMC algorithm for the control of nonlinear processes based on the models identified from input-output data. Models of the form

$$y_k = f(y_{k-1}, y_{k-2}, \dots, y_{k-n_y}, u_{k-1}, u_{k-2}, \dots, u_{k-n_u}) \quad (1)$$

where n_y is the number of past outputs, n_u is the number of past inputs, are considered, whether they are identified using neural networks or polynomial ARMA structure or by some other input-output identification method. y and u are output and input vectors.

A linear model is obtained by linearizing the above nonlinear model at the values of past inputs and outputs at every sampling instant. A minimal state space realization of the linearized model is constructed. Then the linearized model is augmented with the additional linear states to describe the appropriate disturbances and the estimator gains are computed for the augmented system. Once the estimator gains are computed, they are used in order to update the nonlinear states and the augmented linear states to capture the effect of nonlinearity and disturbances. For better understanding, first the procedure for one-step ahead prediction is presented for the linear models and later the nonlinear implementation is outlined. Consider the linear input-output model given by

$$y_j = -A_1 y_{j-1} - A_2 y_{j-2} \dots - A_{n_y} y_{j-n_y} + B_1 u_{j-1} + B_2 u_{j-2} + \dots + B_{n_u} u_{j-n_u} \quad (2)$$

Then a minimal state space realization is constructed using the above model. Software from the package CONSYD is used to construct minimal state space model. The software is based on the algorithm developed in [2] which utilizes Rosenbrock's [13] algorithm. Let the minimal state space model is given by

$$\begin{aligned} x_{j+1} &= \Phi x_j + \Gamma u_j \\ y_j &= C x_j \end{aligned} \quad (3)$$

¹Support for this project was provided by the National Science Foundation (Presidential Young Investigator grant CTS-9057292), the Institute for Systems Research and an unrestricted grant from BP America.

²Author to whom correspondence should be addressed. E-mail: zafiriou@isr.umd.edu

We consider the two sets of augmented models:

Type A:

$$\begin{aligned} x_{j+1} &= \Phi x_j + \Gamma u_j + w_{1j} \\ \eta_{j+1} &= \eta_j + w_{2j} \\ y_j &= Cx_j + \eta_j + v_j \end{aligned} \quad (4)$$

Type B:

$$\begin{aligned} x_{j+1} &= \Phi x_j + \Gamma u_j + \Gamma w_j + w_{1j} \\ w_{j+1} &= w_j + w_{2j} \\ y_j &= Cx_j + v_j \end{aligned} \quad (5)$$

where w_{1j} , w_{2j} and v_j are uncorrelated white noise sequences with $[w_{1j}^T, w_{2j}^T]^T \sim (0, Q)$ and $v \sim (0, R)$, Q and R being covariance matrices associated with process and measurement noise. η is the p -dimensional disturbance vector, x is the n -dimensional state vector, y_j represents the measurement and w is the m -dimensional disturbance vector.

The type A model represents the process model augmented with the disturbance model for disturbances which are step-like at the output. The type B model represents the augmented process and disturbance models for step-like disturbances at the input. Off-set free tracking in the presence of model-plant mismatch can be handled in an effective manner by use of either type of models. Also, the observer designed based on the description of either type can stabilize open-loop unstable processes by putting the closed-loop observer poles inside the unit disk, provided that the controller is designed such that the regulator poles are inside the unit disk. The only technical requirement in using these kinds of disturbance models is that the augmented system is detectable. In general, it is required that the number of new augmented states are less than or equal to the number of outputs for the detectability of the augmented system. This requirement forced us to consider two separate models instead of treating them in a composite setting.

In our development, it is assumed that $Q \approx \begin{bmatrix} \sigma_{w1}^2 & 0 \\ 0 & \sigma_{w2}^2 \end{bmatrix}$ and $R \approx \sigma_v^2 I$ where σ_{w1}^2 , σ_{w2}^2 and σ_v^2 are scalar variances. Define $\sigma_1 = \sigma_{w1}/\sigma_v$, $\sigma_2 = \sigma_{w2}/\sigma_v$ and let $\sigma_v^2 = 1$. The parameters σ_1 and σ_2 are used as a tuning parameters which determine the value of estimator gains. A detailed discussion on this kind of tuning parameter can be found in [5].

Let $K \triangleq \begin{bmatrix} K^1 \\ K^2 \end{bmatrix}$ be the estimator gain. The superscript 1 stands for the gain for the subsystem consisting of original states and 2 stands for the gain for the subsystem consisting of augmented states. These estimator gains are computed by solving an Algebraic Riccati Equation (ARE) [1] using the augmented system matrices and tuning parameters σ_1 and σ_2 .

The one-step ahead prediction equations are given by Type A augmented system:

$$\hat{x}_{j+1|j} = \Phi \hat{x}_{j|j-1} + \Gamma u_j + K^1 [y_j - C \hat{x}_{j|j-1} - \hat{\eta}_{j|j-1}] \quad (6)$$

$$\hat{\eta}_{j+1|j} = \hat{\eta}_{j|j-1} + K^2 [y_j - C \hat{x}_{j|j-1} - \hat{\eta}_{j|j-1}] \quad (7)$$

$$\hat{y}_{j+1|j} = C \hat{x}_{j+1|j} + \hat{\eta}_{j+1|j} \quad (8)$$

By taking the z -transform, $\hat{y}(z)$ is given by

$$\hat{y}(z) = C(zI - \Phi)^{-1} \Gamma u(z) + [C(zI - \Phi)^{-1} K^1 +$$

$$(zI - I)^{-1} K^2] (y(z) - C \hat{x}(z) - \hat{\eta}(z)) \quad (9)$$

Since $C(zI - \Phi)^{-1} \Gamma = (I + \sum_{i=1}^{n_y} A_i z^{-i})^{-1} (\sum_{i=1}^{n_u} B_i z^{-i})$,

$$\begin{aligned} \hat{y}(z) &= [C(zI - \Phi)^{-1} K^1 + (zI - I)^{-1} K^2] (y(z) - \hat{y}(z)) + \\ &(I + \sum_{i=1}^{n_y} A_i z^{-i})^{-1} (\sum_{i=1}^{n_u} B_i z^{-i}) u(z) \end{aligned} \quad (10)$$

$$\begin{aligned} \hat{y}(z) &= - \sum_{i=1}^{n_y} A_i z^{-i} \hat{y}(z) + (\sum_{i=1}^{n_u} B_i z^{-i}) u(z) + (I + \sum_{i=1}^{n_y} A_i z^{-i}) \\ &[C(zI - \Phi)^{-1} K^1 + (zI - I)^{-1} K^2] (y(z) - \hat{y}(z)) \end{aligned} \quad (11)$$

The first two terms on the right hand side of the above prediction equation represent the contribution from the original process model and the third term represents the correction for the assumed disturbance model. Therefore, in the time domain the predicted output at j can be represented as

$$\hat{y}_{j|j-1} = \hat{y}_{j|j-1}^d + \hat{y}_{j|j-1}^s \quad (12)$$

where $\hat{y}_{j|j-1}^d$ represents the deterministic contribution and $\hat{y}_{j|j-1}^s$ represents the correction due to stochastic disturbance assumptions. The deterministic contribution is given by

$$\begin{aligned} \hat{y}_{j|j-1}^d &= -A_1 \hat{y}_{j-1|j-2} - A_2 \hat{y}_{j-2|j-3} \dots - A_{n_y} \hat{y}_{j-n_y|j-n_y-1} + \\ &B_1 u_{j-1} + B_2 u_{j-1} + \dots + B_{n_u} u_{j-n_u} \end{aligned} \quad (13)$$

Stochastic contribution

Define $A_0 \triangleq I$, $\hat{v}(z) \triangleq y(z) - \hat{y}(z)$ and denote the third term on the right hand side of (11) as $\hat{y}^c(z)$.

$$\begin{aligned} \hat{y}^c(z) &= (\sum_{i=0}^{n_y} A_i z^{-i}) C(zI - \Phi)^{-1} K^1 \hat{v}(z) + \\ &(\sum_{i=0}^{n_y} A_i z^{-i}) (zI - I)^{-1} K^2 \hat{v}(z) \end{aligned} \quad (14)$$

Denoting the first term and the second term of the right hand side of the above expression as $\hat{y}^{c1}(z)$ and $\hat{y}^{c2}(z)$ respectively,

$$\hat{y}^c(z) = \hat{y}^{c1}(z) + \hat{y}^{c2}(z) \quad (15)$$

The corresponding time domain expression is

$$\hat{y}_{j|j-1}^c = \hat{y}_{j|j-1}^{c1} + \hat{y}_{j|j-1}^{c2} \quad (16)$$

Now, $\hat{y}^{c1}(z)$ can be represented in the time domain using the state space representation as

$$\begin{aligned} x_{i,l+1} &= \Phi x_{i,l} + K^1 \hat{v}_l \\ y_{i,l} &= A_i C x_{i,l-i} \quad \text{for } i = 1, \dots, n_y \\ \hat{y}_{j|j-1}^{c1} &= \sum_{i=0}^{n_y} y_{i,j} \end{aligned} \quad (17)$$

with initial conditions $x_{i,0} = 0$ for $i = 1, \dots, n_y$ and $\hat{v}_l = 0$ for $l < 0$. Since the identified input-output

model assumes the use of only past n_y outputs information, \hat{y}_{l-i} takes a value of zero for $i > n_y$ or $l-i < 0$. Similarly $\hat{y}_{j|l-1}^{c2}$ is represented in the state space form by replacing Φ by I of appropriate dimension and $A_i C$ by A_i and K^1 by K^2 in (16).

Type B augmented system:

$$\begin{aligned}\hat{x}_{j+1|j} &= \Phi \hat{x}_{j|j-1} + \Gamma [u_j + \hat{w}_{j|j-1}] + K^1 [y_j - C \hat{x}_{j|j-1}] \quad (18) \\ \hat{w}_{j+1|j} &= \hat{w}_{j|j-1} + K^2 [y_j - C \hat{x}_{j|j-1}] \quad (19) \\ \hat{y}_{j+1|j} &= C \hat{x}_{j+1|j} \quad (20)\end{aligned}$$

By taking the z-transform and on simplification

$$\begin{aligned}\hat{y}(z) &= -\sum_{i=1}^{n_y} A_i z^{-i} \hat{y}(z) + \left(\sum_{i=1}^{n_u} B_i z^{-i} \right) (u(z) + w(z)) + \\ &\quad \left(I + \sum_{i=1}^{n_y} A_i z^{-i} \right) [C(zI - \Phi)^{-1} K^1] (y(z) - \hat{y}(z)) \quad (21)\end{aligned}$$

To represent the predicted output in the time domain a similar procedure is used as that for type A augmented system.

2.1. Nonlinear implementation

At sampling instant k , a linear model is obtained by linearizing (1) at $y_{k|k-1}, \dots, y_{k-n_y+1|k-n_y}$ and $u_{k-1}, \dots, u_{k-n_u}$ and is given by

$$\begin{aligned}y_j &= -A_{1,k} y_{j-1} - A_{2,k} y_{j-2} - \dots - A_{n_y,k} y_{j-n_y} + \\ &\quad B_{1,k} u_{j-1} + B_{2,k} u_{j-2} + \dots + B_{n_u,k} u_{j-n_u} \quad (22)\end{aligned}$$

The corresponding minimal state space realization is represented as

$$\begin{aligned}x_{j+1} &= \Phi_k x_j + \Gamma_k u_j \\ y_j &= C_k x_j \quad (23)\end{aligned}$$

Then the states are augmented to obtain type A or type B augmented model and the estimator gain $K_k \triangleq \begin{bmatrix} K_k^1 \\ K_k^2 \end{bmatrix}$ is computed.

Prediction

The predicted output is expressed as the sum of the deterministic contribution and the correction due to stochastic disturbance assumptions.

$$\hat{y}_{k+i|k} = \hat{y}_{k+i|k}^d + \hat{y}_{k+i|k}^c \quad \text{for } i = 1, \dots, P \quad (24)$$

where P is the prediction horizon.

Deterministic contribution

Define, $m(l) \triangleq \min(k, l)$

$$\hat{y}_{k+i|k}^d = f(y_{k+i-1|m(k+i-2)}, \dots, y_{k+i-n_y|m(k+i-n_y-1)}, u_{k+i-1}, \dots, u_{k+i-n_u}) \quad (25)$$

for $i = 1, \dots, P$. The above expression is a nonlinear function in the future manipulated variables. Therefore, to formulate the optimization problem as a single quadratic program the deterministic contribution is subdivided into the effect of past and future.

$$\hat{y}_{k+i|k}^d = \hat{y}_{k+i|k}^{dp} + \hat{y}_{k+i|k}^{df} \quad \text{for } i = 1, \dots, P \quad (26)$$

where $\hat{y}_{k+i|k}^{dp}$ is the effect of past and $\hat{y}_{k+i|k}^{df}$ is the effect of future. The effect of past is computed by setting $u_{k+i} = u_{k-1}$ for $i = 0, 1, \dots, P-1$. The contribution of the effect of future manipulated variables to the predicted output is given as $\sum_{i=1}^l S_{l,k} \Delta u_{k+i-1}$ ($i = 1, 2, \dots, P$), where Δu is the change in manipulated variables, defined as $\Delta u_k \triangleq u_k - u_{k-1}$ and $S_{l,k}$ are the step response coefficient matrices obtained by $S_{l,k} = \sum_{j=1}^l C_k \Phi_k^{j-1} \Gamma_k$ ($l = 1, 2, \dots, P$)

Stochastic contribution

The computation of the linear correction $\hat{y}_{k+i|k}^c$ is exactly the same as that described in the previous subsection with the only modification that the system matrices A_i, Φ, K^P, Γ are replaced by $A_{i,k}, \Phi_k, K_k, \Gamma_k$ respectively. In the absence of measurement information in the future, by taking the conditional mean, it is assumed that $\hat{y}_{k+i} = 0$ for $i = 1, \dots, P$.

Once the predicted output is computed, the future manipulated variables are obtained by solving the optimization problem

$$\min_{\Delta u_k, \dots, \Delta u_{k+M-1}} \sum_{l=1}^P \|\Gamma \hat{y}_{k+l|k} - r_{k+l}\|^2 + \|\Lambda \Delta u_{k+l-1}\|^2 \quad (27)$$

where $\|\bullet\|^2$ is defined by $\|x\|^2 = x^T x$. M is the number of future moves to be optimized. It is assumed that $u_{k+M-1} = u_{k+M} = \dots = u_{k+P-1}$. Γ and Λ are diagonal weight matrices and r is the reference setpoint.

The M future manipulated variables are computed, but only the first move is implemented [4].

2.2. Algorithm schematic

- Linearize (1) at $y_{k|k-1}, \dots, y_{k-n_y+1|k-n_y}$ and $u_{k-1}, \dots, u_{k-n_u}$ to obtain $A_{i,k}$ for $i = 1, \dots, n_y$ and $B_{i,k}$ for $i = 1, \dots, n_u$
- Obtain the minimal state realization of the linearized input-output model
- Compute the step response coefficients
- Compute the estimator gain K_k
- Compute $\hat{y}_{k+i|k}^{dp}$ for $i = 1, \dots, P$ using (25) by setting $u_{k+i} = u_{k-1}$ for $i = 0, 1, \dots, P-1$.
- Compute the linear correction based on the expressions (14)–(16)
- Solve QP and implement u_k
- Obtain $\hat{y}_{k+1|k}$ using (25)

3. Illustration

In this section, the algorithm is applied to control the reactions in series ($A \rightarrow B \rightarrow C$) in a CSTR. The desired product is the intermediate product B. The differential equations describing the system are given by [7]:

$$\begin{aligned}\frac{dx_1}{dt} &= 1 - x_1 - E_3 \exp\left(\frac{-E_1}{x_3}\right) x_1 + E_4 \exp\left(\frac{-E_2}{x_3}\right) x_2 \\ \frac{dx_2}{dt} &= -x_2 + E_3 \exp\left(\frac{-E_1}{x_3}\right) x_1 - E_4 \exp\left(\frac{-E_2}{x_3}\right) x_2 \\ \frac{dx_3}{dt} &= u - x_3 + 0.05(E_3 \exp\left(\frac{-E_1}{x_3}\right) x_1 - E_4 \exp\left(\frac{-E_2}{x_3}\right) x_2) \\ y &= x_2\end{aligned}$$

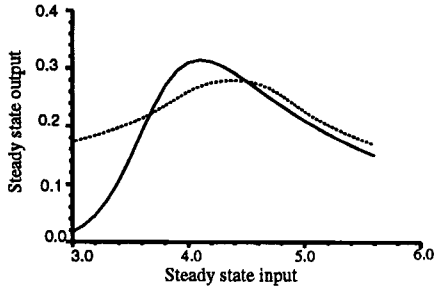


Figure 1: Steady state output vs. Steady state input. Solid line –Plant; Dotted line – Model

with $E_1 = 50$, $E_2 = 70$, $E_3 = 300,000$ and $E_4 = 60,000,000$; where x_1 and x_2 are the dimensionless concentrations of A and B in the reaction mixture, x_3 is the dimensionless reactor temperature and u is the dimensionless temperature of the jacket surrounding the reactor vessel. The reactor equilibrium curve of concentration of B as a function of jacket temperature has a well defined maximum. The control objective is to operate the reactor at the maximum concentration of product B. The concentration has a maximum value of 0.314. The output maximum of this process makes this example a challenging problem for control due to the change in the sign of the gain around this steady state. Gattu and Zafiriou [6] demonstrated the successful application of state estimation NLQDMC algorithm to this process for setpoint changes and input disturbances without model-plant mismatch. In this paper, we use the plant described by the above differential equations and for the model the input-output description taken from [7]. The model-plant mismatch introduced here makes this example more interesting and challenging.

The input-output model is a sigmoid polynomial model and is given as [7]:

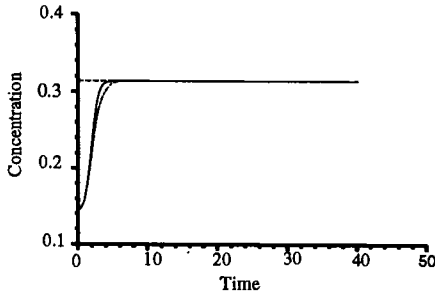


Figure 2: Concentration vs. time. Step setpoint change from left of the peak; Solid line –Type A model; Dash and dotted line – Type B model

$$y(k+1) = \sigma(\theta_0 + \theta_1 y(k) + \theta_2 u_s(k-1) + \theta_3 y^3(k-2) + \theta_4 u_s(k) u_s(k-1) u_s(k-3))$$

where, the map $\sigma(\cdot)$ is the sigmoid function defined as $\sigma(x) = \frac{1}{1+\exp(-x)}$ which is bounded between 0 and 1. The input u_s is the scaled input and is defined as $u_s = \frac{u-2}{4}$. The parameter values are $\theta_0 = -2.751$, $\theta_1 =$

6.791, $\theta_2 = .241$, $\theta_3 = -8.693$ and $\theta_4 = -.226$. The steady state equilibrium curves for the model and plant are given in figure (1). In all the simulations tuning parameter values of $\Gamma = 1$, $P = 10$ and $M = 1$ are used and a sampling time value of 0.5 is used. The input is constrained between -2 and 2 .

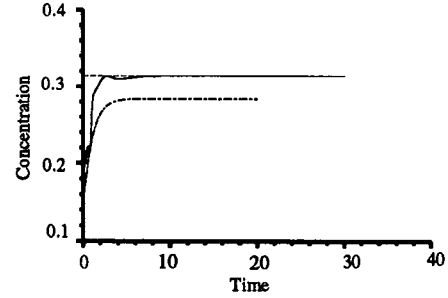


Figure 3: Concentration vs. time. Step setpoint change from right of the peak; Solid line –Type A model; Dash and dotted line – Type B model

Figure 2 demonstrates the response for a step change in setpoint starting from the left of the peak. A step setpoint change is made from the steady state value of $y = 0.1461$ and $u = 3.5$ to a value of $y = 0.314$. The tuning parameter values of $\sigma_1 = 0.0$, $\sigma_2 = 1000.0$ and $\Lambda = 1.0$ are used with both Type A and Type B models. In the entire left side of the peak, the model and plant gains have the same sign. So, it does not pose a challenging problem and the responses using both Type A and Type B models are excellent. Figure 3 demonstrates the response for a step change in setpoint from the right of the peak. A step setpoint change is made from the steady state value of $y = 0.1481$ and $u = 5.5$ to a value of $y = 0.314$. The tuning parameter values of $\sigma_1 = 0.0$, $\sigma_2 = 1000.0$ and $\Lambda = 0.05$ are used with Type A model and the tuning parameter values of $\sigma_1 = 0.0$, $\sigma_2 = 1000.0$ and $\Lambda = 0.3$ are used with Type B model. Note that the gain of the model equilibrium curve changes its sign on the right side of the plant peak. The use of Type B model resulted in steady state offset. The control action got "stuck" at the zero gain area of the model, and the plant settled at an output value corresponding to the input value at the zero gain area. Whereas by using Type A model, the control and observer parameters can be tuned in such a way that it does not get "stuck" at the zero gain area. The oscillations are due to the aggressive control around the zero gain area.

Figure 4 demonstrates the response for a smooth (non-step) setpoint trajectory tracking from the same steady state. With Type B model, tuning parameter values $\sigma_1 = 0.0$, $\sigma_2 = 1000.0$ and $\Lambda = 0.2$ are used. With Type A model, tuning parameter values $\sigma_1 = 0.0$, $\sigma_2 = 1000.0$ and $\Lambda = 0.01$ are used and the change in the manipulated variable is constrained between -0.5 and 0.5 . Similar responses are observed as in the case of step setpoint change. The oscillation in the response when using the Type A model is due to the aggressive control action in the zero gain area. The aggressive control action around the zero gain area is due to the result of using a linear model for future prediction. This situation here points out the shortcoming of the linearization based nonlinear

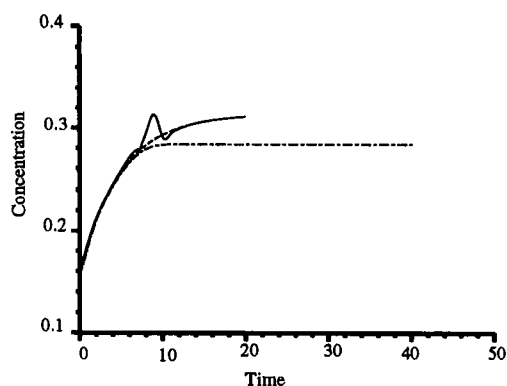


Figure 4: Concentration vs. time. Setpoint trajectory tracking from right of the peak; Solid line –Type A model; Dash and dotted line – Type B model; Dashed line – Setpoint trajectory

MPC algorithms. By choosing a large value of P in an algorithm utilizing nonlinear programming techniques, the oscillations could be reduced.

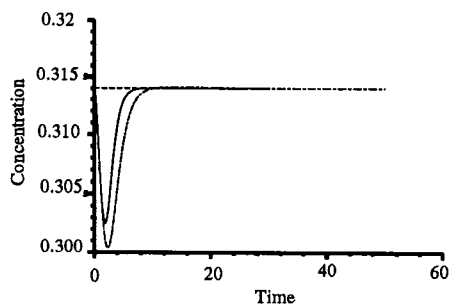


Figure 5: Concentration vs. time. Input disturbances; Solid line –Type A model; Dash and dotted line – Type B model;

Figure 5 demonstrates the response for an input disturbance of -0.2 in a system running at peak. The introduction of negative input disturbance moves the plant curve closer to the zero gain area of the model. The tuning parameter values of $\sigma_2 = 1000.0$ and $\Lambda = 0.5$ are used with Type A model and the tuning parameter values of $\sigma_2 = 1000.0$ and $\Lambda = 0.9$ are used with Type B model. Even in the case of input disturbance, use of Type A model results in a better performance. This is because, the use of a smaller value of Λ with Type B model moves the estimator states into the zero gain area of the model and results in steady state offset.

4. Conclusions

State Estimation NLQDMC algorithm is presented for use with nonlinear input-output models. The proposed algorithm eliminates the major drawbacks of the algorithm presented in [5] for nonlinear state space models. The modifications still preserve the major advantage of the original algorithm: *the computational simplicity by solving only a single quadratic program at each sampling time*

The illustrating example demonstrated the success-

ful application of state estimation NLQDMC for use with input-output models. It was demonstrated that for a system with sign change in the gain, in the presence of model-plant mismatch, the use of augmented models of Type A performs better than augmented models of Type B. The example also demonstrated the shortcomings of the algorithm. It can be seen that the performance loss around the zero gain area is due to the use of linear model for the future prediction.

References

- [1] K. J. Åström and B. Wittenmark, *Computer Controlled Systems: Theory and Design*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
- [2] C. G. Economou, "A New Solution to the Minimal Realization Problem," term project to the Advanced Process Control ChE 770 course at the Univ. of Wisconsin, Madison, 1982.
- [3] C. E. García, "Quadratic/Dynamic Matrix Control of Nonlinear Processes: An Application to a Batch Reaction Process," *AIChE Annual Meeting*, San Francisco, CA, 1984.
- [4] C. E. García and A. M. Morshedi, "Quadratic Programming Solution of Dynamic Matrix Control (QDMC)," *Chem. Eng. Comm.*, **46**, pp. 73-87, 1986.
- [5] G. Gattu and E. Zafiriou, "Nonlinear Quadratic Dynamic Matrix Control With State Estimation," *Ind. and Eng. Chem. Res.*, **31**, pp. 1096-1104, 1992a.
- [6] G. Gattu and E. Zafiriou, "On the stability of Nonlinear Quadratic Dynamic Matrix Control," *3rd IFAC Symposium on Dynamics and Control of Chemical Reactors, Distillation Columns, and Batch Processes (DYCORD+ '92)*, pp. 303-308 of the Preprints, College Park, MD, April 1992b.
- [7] E. Hernandez, "Control of Nonlinear Systems Using Input-Output Information," *Ph.D. Thesis*, Georgia Institute of Technology, 1992.
- [8] E. Hernandez and Y. Arkun, "Study of the control-relevant properties of back propagation neural network models of nonlinear dynamical systems," *Comp. Chem. Eng.*, **15**, pp. 227-240, 1992.
- [9] E. Hernandez and Y. Arkun, "Control of Nonlinear Systems Using Polynomial ARMA models," *AIChEJ*, **39**, pp. 446-460, 1993.
- [10] J. H. Lee and N. L. Ricker, "Extended Kalman Filter Based Nonlinear Model Predictive Control," in *Proceedings of the American Control Conference*, pp. 1795-1799, San Francisco, California, June 1993
- [11] W. C. Li and L. T. Biegler, "Multistep Newton-Type Controller Strategies for Constrained Nonlinear Processes," *Chem. Eng. Res. Des.*, **67**, pp. 562-577, 1989.
- [12] A. A. Patwardhan, J. B. Rawlings and T. F. Edgar, "Nonlinear Model Predictive Control," *Chem. Eng. Comm.*, **87**, pp. 123-141, 1990.
- [13] H. H. Rosenbrock, "Computation of Minimal Representations or a Rational Transfer-Function Matrix," *Proc. IEEE*, **114**, pp. 325-327, 1968.
- [14] J. Saint-Donat, N. Bhat and T. J. McAvoy, "Neural net based model predictive control," *Int. J. Control*, **54**, pp. 1453-1468, 1991.