CONSTRAINED NONLINEAR MODEL PREDICTIVE CONTROL FOR PRACTICAL APPLICATION

A. G. Montandon[1], R. M. Borges[2] , H. M. Henrique[3]

School of Chemical Engineering/Federal University of Uberlandia
Av. João Naves de Ávila, 2160
Phone number: +55-34-3239-4292 Fax number: +55-34-3239-4188
POB 38400-100 Uberlandia - MG, Brazil
E-mails: amanda.montandon@bra.xerox.com.br[1], raquel.borges@ele.ufes.br[2], humberto@ufu.br[3]

Abstract: The proposed MPC is based on a successive linearization of the model at each sampling time and a formulation of a MPC. The cost function of the MPC problem is subject to a reference system as equality constraint and to upper and lower limits in the input variables. In order to satisfy both constraints simultaneously it is needed to include a slack variable in the equality constraint. This slack variable provides more flexibility in the control moves so that the solution of the optimization problem becomes feasible. Proposed controller was implemented to an experimental neutralization pH plant. Results showed a very satisfactory performance of the proposed strategy. *Copyright © 2006 IFAC*

Keywords: Dynamic systems, MPC, Reference System, Constrained nonlinear control, Neural network, pH neutralization process.

1. INTRODUCTION

Process control is concerned to application of automatic control principles for industrial processes. The effect of the global competition in industries has caused the perception of the importance of the product quality in the profitability. Because of this control strategies are used to assure a satisfactory product quality and to decrease raw and energy cost.
However, even though the vast majority of chemical processes are inherently nonlinear, these processes have been controlled by linear controllers. The advantage of this approach is that an easy analytical solution of control problem can be found and a low computational effort is demanded by them. However, the linear approach can be very limiting for highly nonlinear processes and it can lead to unstable solution. The use of nonlinear process models within the control strategy has been shown to provide the potential for significant improvement over linear controllers for nonlinear processes (Bequette, 1991; Henson and Seborg, 1997). Nonlinear model predictive control (NMPC) (Garcia and Morshedi, 1986; Garcia *et al.*, 1989; Gattu and Zafiriou, 1992) and input-output linearizing control (IOLC) are the most widely studied nonlinear control techniques for process control problems. NMPC offers many of the appealing features of linear model predictive control, including explicit compensation for input and output constraints (Meadows et al., 1995). As compared to NMPC, IOLC offers several important advantages including transparent controller tuning and low computational requirements (Kravaris and Kantor, 1990). However, conventional feedback linearization techniques have neither constraint handling (Rawlings et al., 1994) nor predictive capabilities. This has motivated the development of several modifications of the basic input-output linearization approach (Balchen and Sandrib, 1995; Kendi and Doyle, 1995).

On the other hand, the nonlinear approach can result in a large computational effort what turn its use is limited for practical applications. The aim of this work is to present a nonlinear control technique which is computationally feasible for industrial implementation. The proposed strategy is a predictive control technique (MPC) based on a successive linearization of the model via Taylor's series expansion at each sampling time. The cost function of the optimization problem is subject to a first order reference system as a constraint together with upper and lower limits in the inputs. In order to satisfy both constraints simultaneously and to provide a feasible solution, it is necessary to include a slack variable (λ) in the cost function of the optimization problem. The advantage of the proposed algorithm is that it does not need be re-tuned for different operating points. An experimental study was carried out in a pH neutralization plant.

2. THE DYNAMIC SYSTEM

Consider a general dynamic system described by:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x},\mathbf{u}) \qquad \mathbf{y} = \mathbf{h}(\mathbf{x}) \tag{1}$$

Where $\mathbf{u} \in \mathbb{R}^n$ is the vector of manipulated variables, $\mathbf{x} \in \mathbb{R}^m$ is the state vector and $\mathbf{y} \in \mathbb{R}^n$ is the output vector. Linearizing Eq. (1) via Taylor's series expansion around the point immediately earlier to the current point of operation, the following equation is obtained:

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}_{k-1}\mathbf{x} + \mathbf{B}_{k-1}\mathbf{u} + \mathbf{f}_{k-1} \qquad \mathbf{y} = \mathbf{C}_{k-1}\mathbf{x} + \mathbf{h}_{k-1} \quad (2)$$

Where $\mathbf{A}_{k-1} \in \mathbb{R}^{m\times m}$, $\mathbf{B}_{k-1} \in \mathbb{R}^{n\times n}$, $\mathbf{C}_{k-1} \in \mathbb{R}^{n\times m}$, $\mathbf{f}_{k-1}$: $\mathbb{R}^{n+m} \to \mathbb{R}^n$ and $\mathbf{h}_{k-1}$: $\mathbb{R}^m \to \mathbb{R}^n$ are given by:

$$\mathbf{A}_{k-1} = \left(\partial \mathbf{f} / \partial \mathbf{x}\right)_{\mathbf{x}=\mathbf{x}(k-1),\mathbf{u}=\mathbf{u}(k-1)}$$
$$\mathbf{B}_{k-1} = \left(\partial \mathbf{f} / \partial \mathbf{u}\right)_{\mathbf{x}=\mathbf{x}(k-1),\mathbf{u}=\mathbf{u}(k-1)}$$
$$\mathbf{C}_{k-1} = \left(\partial \mathbf{h} / \partial \mathbf{x}\right)_{\mathbf{x}=\mathbf{x}(k-1)}$$
$$\mathbf{f}_{k-1} = \mathbf{f}(\mathbf{x},\mathbf{u})\Big|_{\mathbf{x}=\mathbf{x}(k-1),\mathbf{u}=\mathbf{u}(k-1)}$$
$$\mathbf{h}_{k-1} = \mathbf{h}(\mathbf{x})\Big|_{\mathbf{x}=\mathbf{x}(k-1)}$$

Eq.(2) is integrated from t to t+Δt assuming $\mathbf{u}(k)$ constant during sampling instant:

$$\begin{aligned} \hat{\mathbf{x}}(k+1) &= \boldsymbol{\Phi}\,\mathbf{x}(k) + \boldsymbol{\Psi}\,\mathbf{u}(k) + \boldsymbol{\Omega}\mathbf{f}_{k-1} \\ \hat{\mathbf{y}}(k) &= \mathbf{C}\,\mathbf{x}(k) + \mathbf{h}_{k-1} \end{aligned} \quad (3)$$

Where $\boldsymbol{\Phi} \in \mathbb{R}^{m\times m}$, $\boldsymbol{\Psi} \in \mathbb{R}^{m\times n}$, $\boldsymbol{\Omega} \in \mathbb{R}^{m\times m}$ are:

$$\boldsymbol{\Phi} = e^{\mathbf{A}\Delta t},\ \boldsymbol{\Psi} = \mathbf{A}^{-1}\left(e^{\mathbf{A}\Delta t} - \mathbf{I}\right)\mathbf{B},\ \boldsymbol{\Omega} = \mathbf{A}^{-1}\left(e^{\mathbf{A}\Delta t} - \mathbf{I}\right)$$

Since $\mathbf{A}$ is nonsingular. Eq.(3) can be written for each prediction instant from k = 1 to k = P where P is a prediction horizon and M control horizon with $P \geq M$ and $\Delta\mathbf{u}(k + j) = \mathbf{0}$ to $M < j \leq P$. The resulting set of equations can be put in a matrix form:

$$\hat{\mathbf{y}} = \boldsymbol{\Gamma} \cdot \Delta\mathbf{u} + \boldsymbol{\gamma} \quad (4)$$

Where:

$$\hat{\mathbf{y}} = \left[\hat{\mathbf{y}}(k+1) \quad \hat{\mathbf{y}}(k+2) \quad \cdots \quad \hat{\mathbf{y}}(k+M) \quad \cdots \quad \hat{\mathbf{y}}(k+P)\right]^T$$
$$\Delta\mathbf{u} = \left[\Delta\mathbf{u}(k) \quad \Delta\mathbf{u}(k+1) \quad \cdots \quad \Delta\mathbf{u}(k+M)\right]^T$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \mathbf{C}\boldsymbol{\Psi} & [\mathbf{0}] & \cdots & [\mathbf{0}] \\ \mathbf{C}\,(\boldsymbol{\Phi}+\mathbf{I})\boldsymbol{\Psi} & \mathbf{C}\boldsymbol{\Psi} & \cdots & [\mathbf{0}] \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{C}\left(\sum_{i=1}^{M}\boldsymbol{\Phi}^{i-1}\right)\boldsymbol{\Psi} & \mathbf{C}\left(\sum_{i=1}^{M-1}\boldsymbol{\Phi}^{i-1}\right)\boldsymbol{\Psi} & \cdots & \mathbf{C}\boldsymbol{\Psi} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{C}\left(\sum_{i=1}^{P}\boldsymbol{\Phi}^{i-1}\right)\boldsymbol{\Psi} & \mathbf{C}\left(\sum_{i=1}^{P-1}\boldsymbol{\Phi}^{i-1}\right)\boldsymbol{\Psi} & \cdots & \mathbf{C}\left(\sum_{i=1}^{P-M}\boldsymbol{\Phi}^{i-1}\right)\boldsymbol{\Psi} \end{bmatrix}$$

$$\boldsymbol{\gamma} = \begin{bmatrix} \mathbf{y}(k) + \mathbf{C}\boldsymbol{\Phi}\Delta\mathbf{x}(k) \\ \mathbf{y}(k) + \mathbf{C}\left(\sum_{i=1}^{2}\boldsymbol{\Phi}^{i}\right)\Delta\mathbf{x}(k) \\ \vdots \\ \mathbf{y}(k) + \mathbf{C}\left(\sum_{i=1}^{M}\boldsymbol{\Phi}^{i}\right)\Delta\mathbf{x}(k) \\ \vdots \\ \mathbf{y}(k) + \mathbf{C}\left(\sum_{i=1}^{P}\boldsymbol{\Phi}^{i}\right)\Delta\mathbf{x}(k) \end{bmatrix}$$

2.1 The Reference System

The controller is designed in order to transform the closed loop system in a first order system:

$$\frac{d\mathbf{y}^*}{dt} = \mathbf{K}\left(\mathbf{y}^{SP} - \mathbf{y}\right) \quad (5)$$

Where $\mathbf{K} \in \mathbb{R}^{n\times n}$ is tuning parameter, $\mathbf{y}^{SP}$ are the setpoints and $\mathbf{y}^*$ are the reference output trajectories. The controller must satisfy the following reference system:

$$\frac{d\mathbf{y}}{dt} = \frac{d\mathbf{y}^*}{dt} \quad (6)$$

Infeasible solutions of the MPC optimization problem can occur when equality constraints represented by Eq.(6) and hard constraints (upper and lower bounds for inputs) must be satisfied simultaneously. To overcome this problem, a slack variable ($\boldsymbol{\lambda}$) is introduced into Eq.(6) in order to allow the system to deviate from reference system and to satisfy the hard constraints. Therefore, introducing slack variable $\boldsymbol{\lambda} \in \mathbb{R}^n$ into Eq. (6) and substituting Eq. (2) into Eq. (6) we obtain:

$$\mathbf{C}\,\mathbf{f}_{k-1} + \mathbf{C}\mathbf{A}_{k-1}\mathbf{x}(k) + \mathbf{C}\mathbf{B}_{k-1}\mathbf{u}(k) + \boldsymbol{\lambda}(k) = \mathbf{K}\left(\mathbf{y}^{SP}(k) - \mathbf{y}(k)\right) \quad (7)$$

Eq. (7) can be written for each prediction instant and it can be put in a matrix form:

$$\mathbf{D}\ \mathbf{z} = \mathbf{b} \quad (8)$$

Where:

$$\mathbf{D} = \begin{bmatrix} \mathbf{CB} & [\mathbf{0}] & \cdots & [\mathbf{0}] & [\mathbf{I}] & [\mathbf{0}] & [\mathbf{0}] & [\mathbf{0}] & \cdots & [\mathbf{0}] \\ (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Psi} & \mathbf{CB} & \cdots & [\mathbf{0}] & [-\mathbf{I}] & [\mathbf{I}] & [\mathbf{0}] & [\mathbf{0}] & \cdots & [\mathbf{0}] \\ (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}\mathbf{C} & (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Psi} & \cdots & [\mathbf{0}] & [\mathbf{0}] & [-\mathbf{I}] & [\mathbf{I}] & [\mathbf{0}] & \cdots & [\mathbf{0}] \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}^{P-1}\boldsymbol{\Psi} & (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}^{P-2}\boldsymbol{\Psi} & \cdots & (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}^{P-M-1}\boldsymbol{\Psi} & [\mathbf{0}] & [\mathbf{0}] & [\mathbf{0}] & [\mathbf{0}] & \cdots & [\mathbf{I}] \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} \mathbf{K}\mathbf{y}^{SP}(k) - (\mathbf{CA}+\mathbf{KC})\mathbf{x}(k) - \mathbf{C}\mathbf{f}_{k-1} \\ \mathbf{K}\Delta\mathbf{y}^{SP}(k+1) - (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}\mathbf{x}(k) \\ \mathbf{K}\Delta\mathbf{y}^{SP}(k+2) - (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}^2\mathbf{x}(k) \\ \vdots \\ \mathbf{K}\Delta\mathbf{y}^{SP}(k+P) - (\mathbf{CA}+\mathbf{KC})\boldsymbol{\Phi}^P\mathbf{x}(k) \end{bmatrix}$$

$$\mathbf{z} = \left[\Delta\mathbf{u}(k) \quad \cdots \quad \Delta\mathbf{u}(k+M) \quad \boldsymbol{\lambda}(k) \quad \cdots \quad \boldsymbol{\lambda}(k+P)\right]^T \tag{9}$$

2.2 The proposed MPC Design

The cost function of the MPC problem is defined as:

$$\min_{\Delta\mathbf{u}(k),\ldots,\Delta\mathbf{u}(k+M),\,\boldsymbol{\lambda}(k),\ldots,\,\boldsymbol{\lambda}(k+P)} \mathbf{J} = \frac{1}{2}\left(\Delta\mathbf{u}^T\mathbf{R}\,\Delta\mathbf{u} + \boldsymbol{\lambda}^T\mathbf{S}\boldsymbol{\lambda}\right) \tag{10}$$

and also reorganized as a cost function of a quadratic programming problem:

$$\begin{aligned} &\min_{\mathbf{z}(k),\ldots,\mathbf{z}(k+M+P+2)} \ \mathbf{J} = \frac{1}{2}\mathbf{z}^T\mathbf{H}\,\mathbf{z} \\ &\text{subject to:} \\ &\mathbf{D}\,\mathbf{z} = \mathbf{b} \\ &\mathbf{u}_{min}(k+j) \le \mathbf{u}(k+j) \le \mathbf{u}_{max}(k+j) \\ &-\left|\Delta\mathbf{u}_{max}(k+j)\right| \le \Delta\mathbf{u}(k+j) \le \left|\Delta\mathbf{u}_{max}(k+j)\right| \\ &j = 0,\cdots,P \end{aligned} \tag{11}$$

Where $\mathbf{z}$ is given by Eq.(9) and matrix $\mathbf{H} \in \mathbb{R}^{(P+M+2)\times(P+M+2)}$ is:

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & [\mathbf{0}] \\ [\mathbf{0}] & \mathbf{S} \end{bmatrix}$$

Borges (2001) showed the proposed controller as presented here can not eliminate offset. Borges (2001) unless that the following change in Eq. (8) is made:

$$\mathbf{f}_{k-1} \approx \frac{\mathbf{y}_k - \mathbf{y}_{k-1}}{\Delta t} \tag{12}$$

Where $\mathbf{y}_k$ and $\mathbf{y}_{k-1}$ are the system outputs at k^{th} and $(k-1)^{th}$ sampling instants respectively. Both vectors are available at k^{th} sampling instant.

2.3 The MPC design

A classical MPC is designed in order to compare results. The following classical MPC is defined:

$$\begin{aligned} &\min_{\Delta\mathbf{u}(k),\ldots,\,\Delta\mathbf{u}(k+M)} \mathrm{J} = \frac{1}{2}\left(\mathbf{e}^T\mathbf{Q}\mathbf{e} + \Delta\mathbf{u}^T\mathbf{R}\Delta\mathbf{u}\right) \\ &\mathbf{u}_{min}(k+j) \le \mathbf{u}(k+j) \le \mathbf{u}_{max}(k+j) \quad (j = 0,\ldots,M) \\ &-\left|\Delta\mathbf{u}_{max}(k+j)\right| \le \Delta\mathbf{u}(k+j) \le \left|\Delta\mathbf{u}_{max}(k+j)\right| \\ &j = 0,\cdots,P \end{aligned} \tag{13}$$

Where $\mathbf{e}$ represents the setpoint deviation vector and it is given by $\mathbf{e} = -\boldsymbol{\Gamma}\Delta\mathbf{u} + \mathbf{e}'$ where $\mathbf{e}' = (\mathbf{y}^{SP} - \boldsymbol{\gamma})$. The optimization problem represented by Eq. (13) can also be put as a quadratic programming problem.

3. EXPERIMENTAL APPLICATION

3.1 The system

Consider a neutralization process that occurs in the CSTR shown in Fig.1. The system involves the dynamic behavior of pH and contains two inputs, an acid stream (HNO_3), Q_1, a base stream (NaOH), Q_2, and a single output, pH. The liquid level is constant and the chemical reactions involved are:

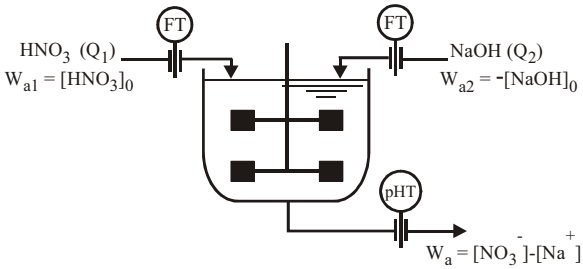

Fig. 1. Experimental setup

$$\begin{aligned} &H_2O \Leftrightarrow OH^- + H^+ \\ &H_2CO_3 \Leftrightarrow HCO_3^- + H^+ \\ &HCO_3^- \Leftrightarrow CO_3^= + H^+ \end{aligned} \tag{14}$$

The dynamic of the process is given by the following physical model (Montandon, 2005):

$$V\frac{d(W_a)}{dt} = Q_1(W_{a1} - W_a) + Q_2(W_{a2} - W_a) \tag{15}$$

$$W_a(0) = \overline{W}_a \tag{16}$$

where $W_a = [NO_3^-] - [Na^+]$ is a reaction invariant because it is not affect by extension of the reaction. Eq.(16) represents initial condition for the reactor:

$$\overline{W}_a = \frac{(Q_1 W_{a1} + Q_2 W_{a2})}{(Q_1 + Q_2)} \tag{17}$$

$W_a(t)$ can be obtained by integrating of Eq.(15), $[H^+]$ is obtained by substituting $W_a(t)$ into Eq.(18) and pH(t) is obtained by substituting $[H^+]$ into Eq. (19):

$$W_a = [H^+] - \frac{K_W}{[H^+]} \tag{18}$$

$$pH = -\log[H^+] \tag{19}$$

Table 1 gives the nominal values of system parameters. This system is interesting from control point of view because it is strongly nonlinear as it can be seen in Fig. 2.

Table 1. Nominal values of the system parameters.

Variable	Symbol	Nominal values
Volume	V	4459,94 cm3
Acid flowrate	Q_1	12,0 mL/s
Base flowrate	Q_2	12,0 mL/s
pH	pH	7,0
Acid conc. in Q_1	$[HNO3]_0$	3,611e-03 M
Base conc. in Q_2	$[NaOH]_0$	3,611e-03 M
W_a in Q_1	W_{a1}	3,611e-03 M
W_a in Q_2	W_{a2}	-3,611e-03 M
W_a in output	W_a	0,0

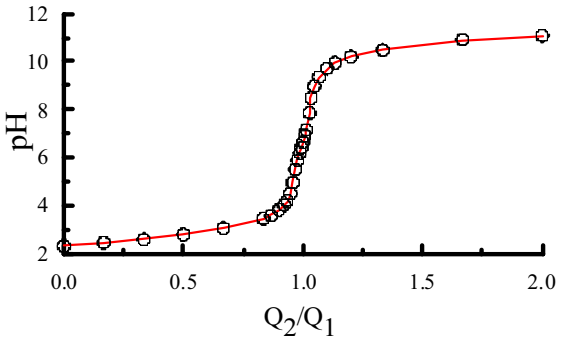

Fig. 2. Experimental Titration curve.

3.2 The neural network model

The problem of using physical model is that it can be quite difficult to get or insufficiently accurate to be used in a MPC. In addition, input/output data are generally available in industrial application. To take advantage of this fact, it was developed a version of proposed controlled based on a neural network models. Before introducing NN models, it is convenient to discuss some practical aspects of the representation of these types of dynamic systems. The natural way to represent the dynamic system represented by Eq. 1 is to use neural networks with neurons with dynamic characteristics (You and Nikolaou, l993). This approach has the advantage of producing models of small dimension. For a single input single output system (SISO), the resulting neural network will have only one input. The main disadvantage of these types of neural networks is the training phase. It is very time consuming and hard to converge. A popular alternative is to consider a neural network with static neurons representing a discrete approximation of the dynamic system in the form of a NARX model (Su *et al*, l992). In this case, the main advantage is associated with the simplicity of the training phase. The disadvantage is that the number of required network inputs increases with input and output lags causing a huge increase in network structure. Another problem is that the determination of the input and output lags requires very often a tedious iterative process. For these reasons, in this work we consider a different alternative that consists of the direct representation of Eq.(1) with a static neural network followed by a numerical integration to recover **y**(t+1).

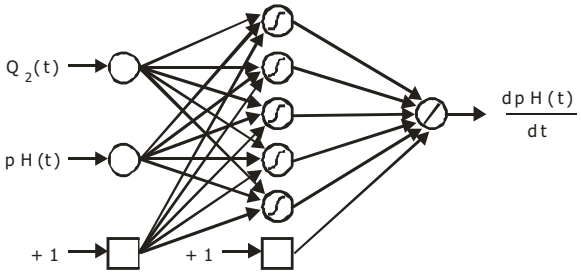

Fig. 3. Neural network topology used.

The Fig. 3 shows the neural network topology schematically. If the value of **y**(t) from plant is used as initial condition for obtaining **y**(t+1) by integration then an one step ahead prediction is obtained, but if the value of **y**(t) is obtained from previous integration step then a multiple step ahead prediction is obtained. For pH neutralization process, the FNN model predicts the time derivatives of pH as a function of the base flow rate and the system pH. The system was excited using a random uniform step sequence for base flow rate Q_2 with a step probability (probability of a step change occurring at any given sampling instant) equal to 0.8 (Bomberger and Seborg, 1997). Q_1 was kept constant in its nominal value. Fig. 4 shows the experimental input and output and the one step ahead prediction performed by the neural model. A sampling period of 10s was used. Data from the time interval [0 to 256 min] were used to the FNN training phase and from the interval [256 to 400 min] to the validation phase. The network inputs were $Q_3(t)$, pH(t) and the network output was d[pH(t)]/dt. The derivatives of pH were calculated numerically by finite difference schemes of filtered pH values. After a cross validation procedure a neural network with five nonlinear hidden neurons (hyperbolic tangent activation function) and one linear output neuron was selected and trained until convergence using the Levenberg-Marquardt method.

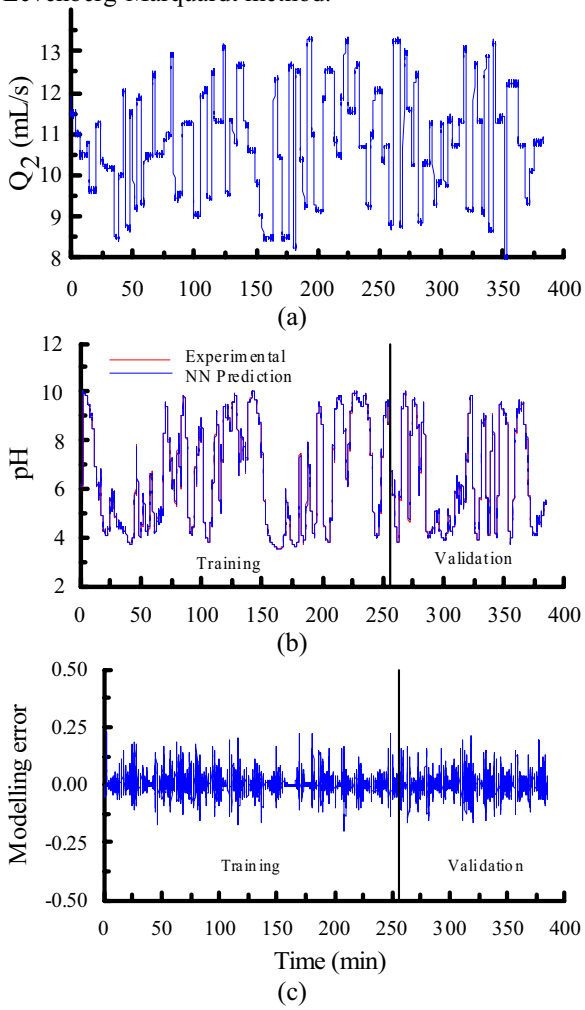

Fig. 4. Experimental data: a). Input sequence for Q_2. b). System output and one step ahead NN prediction. c). Modelling error using NN model.

Fig. 4 shows the data used for training and validation phases and the NN prediction. Prediction results by using physical model were not accurate and this

model was not used in MPC calculation. These results were not shown here to save space. In order to illustrate the advantage of proposed technique over classical PID and MPC technique, a digital PID and a constrained predictive controller (Eq. 13) were implemented experimentally and compared to the proposed controller (Eq. 11) for servo and regulator problems (unmeasured perturbation). All controllers were first tuned by simulation tests using the NN model to represent the real system. In following field tuning were performed for three controllers using preliminary parameters in order to get the best controller parameters experimentally adjusted. The tuned PID parameters are $\Delta t = 10$ s, $K_c = 0.5$ s/mL, $\tau_I = 90$ s and $\tau_D = 0$ s. For classical MPC the tuned parameters are $\Delta t = 10$ s, M = 10, P = 20, Q = 1, R = 150, $|\Delta Q_{3\ max}| = 1$ mL/s, $Q_{3\ max} = 30$ mL/s and $Q_{3\ min} = 5$ mL/s. For proposed MPC the tuned parameters are $\Delta t = 10$ s, K=5e-03 s^{-1}, M = 10, P = 20, R = 1, S = 100, $|\Delta Q_{3\ max}| = 1$ mL/s, $Q_{3\ max} = 30$ mL/s, $Q_{3\ min} = 5$ mL/s. Fig. 5 and Fig. 6 show results of the controllers in servo problems.

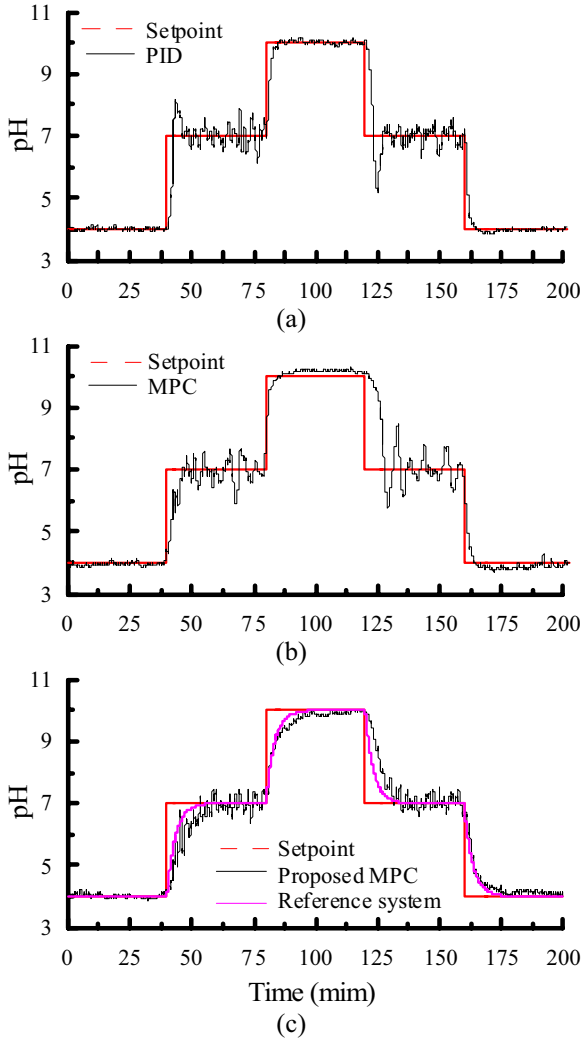

Fig. 5. Closed loop responses for setpoint changes: a). PID, b). Classical MPC, c). Proposed MPC.

Results from Fig. 5 reveal that the proposed MPC yielded a symmetric response for setpoint changes. This result is a consequence of the system reference put as an equality constraint in the quadratic programming of the control problem. As a consequence of this the closed loop response of the real system is basically a first order response. Therefore, controller re-tuning was not needed when new operation points are required. On the other hand, the Fig. 5 shows also that the PID and classical MPC performance are acceptable for system operation at low and high pH values, but the responses deteriorated considerably for system operation around of pH = 7. Consequently, a different set of the controller parameters must be required for good performance at different operational conditions. This is clearly not a desirable situation in any application since it greatly increases the maintenance needs of the controller. Fig. 6 shows the control actions for the three controllers. This Figure reveals that the PID yielded more aggressive control moves than MPC approaches. The more conservative control moves obtained by MPC approaches are because of the presence of hard constraints in the manipulated variable.

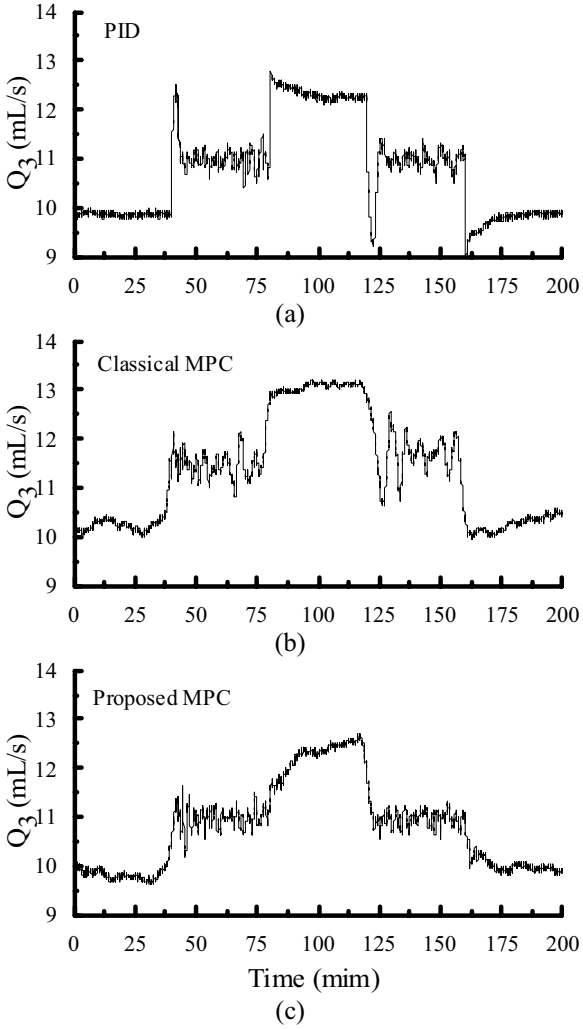

Fig. 6. Control actions: a). PID, b). Classical MPC, c). Proposed MPC.

Next, the capacity of unmeasured perturbation rejection of the controllers was tested. Acid flowrate was chosen as system load and the controller parameters were kept the same for all controllers. The run was started with $Q_1 = 12$ mL/s, it was changed to $Q_1 = 13.5$ mL/s at instant t = 40 min, to $Q_1 = 12$ mL/s

at instant t = 80 min, to Q_1 = 10.5 mL/s at instant t = 120 min and to Q_1 = 12 mL/s at instant t = 160 min. After t = 160 min the acid flowrate was kept in its initial value Q_1 = 12 mL/s. The initial condition of the system is pH =7. In this region the magnitude of the change in Q_1 are too severe because of the high value of the system static gain. Fig. 7 shows the controller performances for load changes.

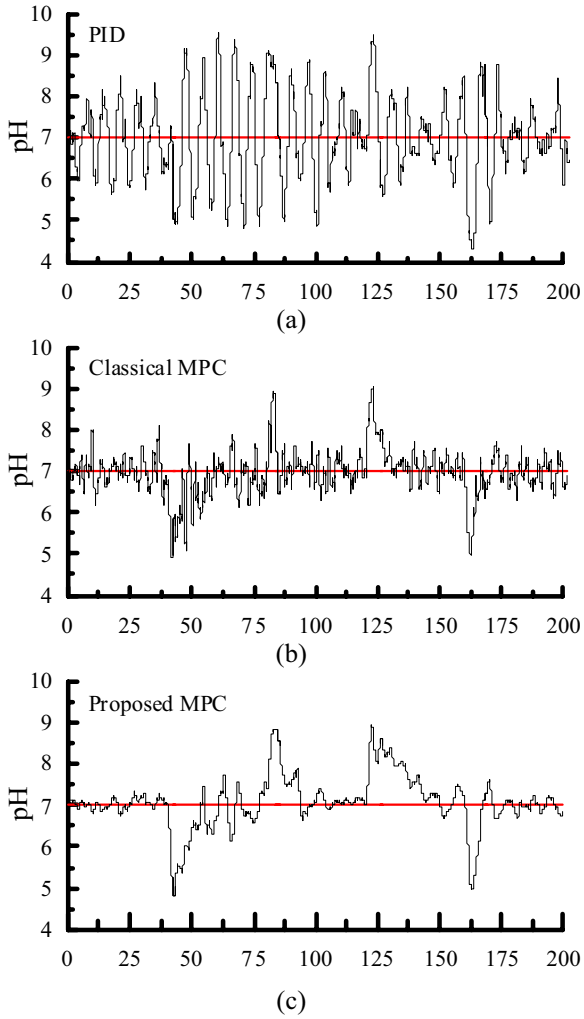

Fig. 7. Closed loop responses for load changes: a). PID, b). Classical MPC, c). Proposed MPC.

Fig. 7 reveals that the PID controller yielded unstable response, very oscillatory with increasing amplitudes. The MPC approaches yielded stable and acceptable responses around of the operation point (pH = 7). These behaviors are remarkable because of the NN network was just trained to Q_1 = 12 mL/s. Due to high sensitivity of the static gain of the system around of pH =7 (see Fig. 2) these changes in Q_1 have a drastic impact in accuracy of the NN model. In spite of this the MPCs controlled the system in a stable fashion. The PID control moves were very aggressive and the MPC control moves were acceptable and no violation of the limits occurred. Results for control moves were not shown here because of slack of space.

4. CONCLUSION

This paper introduces a new MPC strategy based on a first order reference system. The proposed control algorithm was developed and implemented experimentally. Results of a pH neutralization process showed the proposed controller was clearly superior to PID and classical MPC for servo and regulator problem. The proposed method retains the computational simplicity while providing some desirable features from model predictive control, such as constraint handling, incorporating future setpoint changes, penalizing large control move increments by selecting appropriate weighting parameters in the objective function. It was also verified the proposed technique yielded promising results in a real control problem confirming its good potential for practical implementation due to low computational requirements, good closed loop performance as well as transparent controller tuning.

REFERENCES

Balchen, J. G. and Sandrib, B. (1995). Input saturation in nonlinear multivariable processes resolved by nonlinear decoupling. *Model. Ident. Control*, **Volume 16**, 95-106.

Bequette, B. W. (1991). Nonlinear Control of Chemical Process: A Review. *Ind. Eng. Chem. Res,* **Volume 30**, 1391-1413.

Borges, R. M. (2001). Controle Preditivo Basedo em Sistema de Referência. *M. Sc. Thesis*. Uberlandia, Brazil. *(in Portuguese)*

Garcia, C. E., Morshedi, A.M. (1986). Quadratic Programming Solution of Dynamic Matrix Control (QDMC). *Chem. Engng.Commun*, **Volume 46**, 73-87.

Garcia, C.E. , Prett, D.M. and Morari, M. (1989). Model Predictive Control – A Survey. Automatica **Volume 25**, 335-348.

Gattu, G. and Zafiriou, E. (1992). Nonlinear Quadratic Dynamic Matrix Control With State Estimation. *Ind. and Eng. Chem. Res,* **Volume 31**, 1096-1104.

Henson, M. A., Seborg, D. E.(1997). *Nonlinear Process Control*. Prentice Hall, New Jersey.

Kendi, T. A. and Doyle, F. J.(1995). An anti-windup scheme for input-output linearization. *Proc. European Control Conf*., Rome, Italy.

Kravaris, C. and Kantor, J. C. (1990). Geometric methods for nonlinear process control: 2. controller synthesis. *Ind. Eng. Chem. Res*., **Volume 29**, 2310-2323.

Meadows, E. S., Henson M. A., Eaton, J. W. and Rawlings, J. B. (1995). Recending horizon control and discontinuous state feedback stabilization. *Int. J. Control*, **Volume 62**, 1217-1229.

Montandon, A. G. (2005). Controle Preditivo em Tempo Real com Trajetória de Referência Baseado em Modelo Neural para Reator de Neutralização. *M. Sc. Thesis*. Uberlandia, Brazil. *(in Portuguese)*

Rawlings, J. B., Meadows, E. S. and Muske, K. R. (1994). Nonlinear model predictive control: A tutorial and survey. *Proc. IFAC Symposium on Advanced Control of Chemical Processes*, Kyoto, Japan, pp. 203-214.