MIMO Model Predictive Control with Local Linear Models

JAKUB NOVÁK, PETR CHALUPA, VLADIMÍR BOBÁL
Faculty of Applied Informatics
Tomas Bata University in Zlin
Nam. T.G.Masaryka 5555, 76001, Zlin
CZECH REPUBLIC
jnovakt@fai.utb.cz http://www.fai.utb.cz

Abstract: - The performance of a Model Predictive Control (MPC) algorithm depends on the quality of the derived model. Using a divide-and-conquer strategy process operations were partitioned into several operating regions and within each region, a local linear model was developed to model the process. This set of locally linearized models was simply and effectively combined into a global description of a multivariable nonlinear plant. To save on computational load, a linear model was obtained by interpolating these linear models at each sample point and then this linearized model was used in a Generalized Predictive Control (GPC) framework to calculate the future behavior of the process. Thus, time-consuming nonlinear quadratic optimization calculations, which are normally necessary in nonlinear predictive control, can be avoided. Modeling and controller design procedure was demonstrated using a simulated pH neutralization process with two inputs and two outputs.

Key-Words: - Predictive control, pH neutralization, local model networks, multiple models, linearization

1 Introduction

Model predictive control (MPC) [1] has been a major research topic for the last 30 years. The reason for this is the ability of MPC to optimally control multivariable system under various constraints. The main idea of the MPC is to calculate the actual and the subsequent control signals by minimizing the quadratic deviation of a reference signal and an output signal in a given future horizon. The solution to this optimization problem is the optimal input signal to the system at that particular time [2]. According to the receding horizon control strategy, only the first control signal is used at the process input, and in the next sampling point the procedure is repeated. Conventional MPC techniques are based on the use of linear models. Linear MPCs can yield a satisfactory performance if the process is reasonably linear, or is operated close to the nominal steady state. However, a linear model is not sufficient to capture the properties of chemical engineering processes.

The poor performance of linear MPCs for processes with a strong degree of nonlinearity (for example pH control or batch reactors) has motivated the development of nonlinear model predictive control (NMPC), where a more accurate (nonlinear) model of a plant is used for prediction and optimization. Qin and Badgwell [3] presented a survey of nonlinear model predictive control applications in industry. In NMPC, the importance of having an accurate process model is crucial, and several nonlinear models that have been utilized for NMPC can be found in the literature. In Ref. [4], a Wiener-type nonlinear black box model was developed for capturing the dynamics of open loop stable Multiple Input Multiple Output (MIMO) nonlinear systems with deterministic inputs. Multiple model approaches to modeling and control have become an attractive research field in recent years [5]. A multimodel approach has advantages in controlling industrial processes, especially those with inherent nonlinearity, a wide operating range, or load disturbances. Based on a divide-and-conquer strategy, multimodel approaches can be used to develop local linear models or controllers corresponding to typical operating regimes. Galan et al. [6] reported the real-time implementation of a multilinear model based control strategies for a bench top–scale pH neutralization reactor. Xue and Li have proposed a multiple model predictive control (MMPC) strategy based on the Takagi–Sugeno (T–S) model in [7]. In their approach different predictive controllers were designed for different local models with different local constraints.

The pH neutralization process was chosen as a benchmark for control algorithms in several studies as it exhibits significant nonlinear behavior.

ISBN: 978-1-61804-004-6

2 Local Linear Models

Modeling nonlinear dynamic systems from observed data and a priori engineering knowledge is a major area of science and engineering. In recent years a great deal of work has appeared in new areas, such as fuzzy modeling and neural networks. LMNs were first introduced by Johansen and Foss [8] to describe a set of submodels, each of which was valid for a specific regime in an operating space, weighted by activation function. A LMN is a generalization of a radial basis function (RBF) network, in which individual neurons are replaced by local submodels with basic functions defining the regions of validity of individual submodels, according to the expected operating regions of a plant.

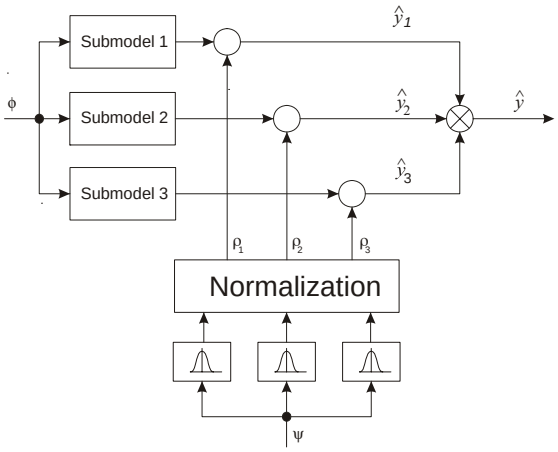

Figure 1. A local model network scheme

The LMN output is given by:

$$\hat{y}(k) = \sum_{i=1}^{M} \rho_i(\boldsymbol{\psi}(k))\hat{y}_i(k) \qquad (1)$$

where $\boldsymbol{\psi}(k)$ is a vector of scheduling variables, $\rho_i(k)$ is a normalized validity function, and $\hat{y}_i(k)$ is the output of the *i*-th model. The network that is described by Equation (1) is shown in Figure 2. The blending of local models is calculated using weighting or validity functions. Although any function with a locally limited activation may be applied as a validity function, a common choice for this function takes the Gaussian form. The validity function for the *i*-th model is given by:

$$\tilde{\rho}_i(\boldsymbol{\psi}(k)) = \exp\left(-\frac{1}{2}\left(\boldsymbol{\psi} - c_i\right)^T \sigma_i^{-2}\left(\boldsymbol{\psi} - c_i\right)\right) \qquad (2)$$

where the parameters c_i, σ_i , define the Gaussian centre and width, respectively, and the scheduling variable ψ can be a system state or any system variable.

Basically, there are two ways to design controllers for local model structures: the linearization-based and local model-based approaches. In the linearization-based approach, the local model network is linearized at the current operating point, and the linear controller is designed. The linearization of the LMN is very simple due to the structure of the model. A linear model is obtained by interpolating these linear models at each sample point. In the second approach a local controller is designed for each local model, and the control output is then calculated as an interpolation of the local controller outputs according to the current operating point.

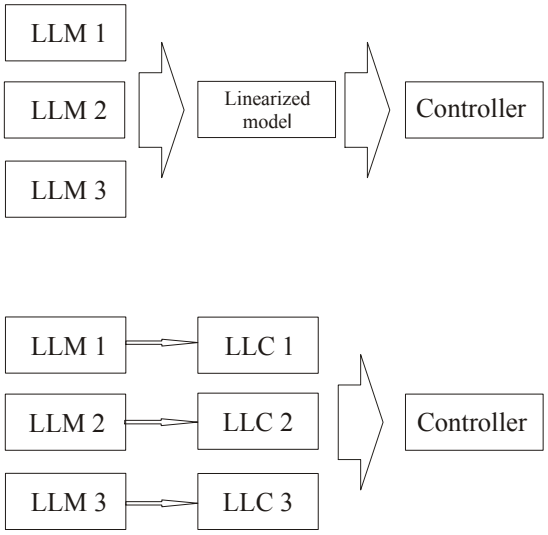

Figure 2. Controller design using linearization and local models (LLM = local linear model, LLC = local linear controller)

3 pH Neutralization

The system chosen for this study is a pH neutralization process. The process was modeled using a nonlinear first-principles model, which was computationally too demanding for MPC computations. Therefore it is a good benchmark example for local modeling and control. Based on the time constants of the process the sampling period was chosen to be 15 s. The process consists of an acid (HNO3) stream, a buffer (NaHCO3) stream and a base (NaOH) stream being continually mixed in the tank (Figure 3). The model is based on the assumptions that the streams are perfectly mixed, and the density is constant throughout the entire tank. The process is aimed at controlling the

ISBN: 978-1-61804-004-6

pH value of the outlet stream and the level *h*, by varying the inlet base and acid streams, Q_3 and Q_1, respectively. The outlet flow rate is dependent on the fluid height in the tank as well as the position of the valve. A differential equation that describes the total mass balance of the tank is

$$\frac{dh}{dt}=\frac{1}{A}\left(Q_1+Q_2+Q_3-c\sqrt{h}\right) \tag{3}$$

where *c* is a valve constant, *A* is the tank cross-sectional area, and *h* is the tank level.

The differential equations for the effluent reaction invariants and can be derived as

$$\begin{aligned}\frac{dW_a}{dt}&=\frac{1}{Ah}\begin{pmatrix}Q_1(W_{a1}-W_a)+Q_2(W_{a2}-W_a)\\+Q_3(W_{a3}-W_a)\end{pmatrix}\\ \frac{dW_b}{dt}&=\frac{1}{Ah}\begin{pmatrix}Q_1(W_{b1}-W_b)+Q_2(W_{b2}-W_b)\\+Q_3(W_{b3}-W_b)\end{pmatrix}\end{aligned} \tag{4}$$

where W_{ai} and W_{bi} are the chemical reaction invariants of the *i*-th stream. The variables are defined in Table 1.

Table 1 Parameters of the pH neutralization plant

Symb.	Variable	Nom. value
A	Tank area	$207\ cm^2$
h	Tank level	*14 cm*
Q_1	Acid flow rate	*16.6 ml/s*
Q_2	Buffer flow rate	*0.55 ml/s*
Q_3	Base flow rate	*15.6 ml/s*
c	Valve constant	$8\ ml/s\sqrt{cm}$
W_{a1}	$[HNO_3]_1$	*0.003 mol*
W_{a2}	$-[NaHCO_3]_2$	*-0.03 mol*
W_{a3}	$-[NaHCO_3]_3-[NaOH]_3$	*-0.00305 mol*
W_{b1}	$[NaHCO_3]_1$	*0 mol*
W_{b2}	$[NaHCO_3]_2$	*0.03 mol*
W_{b3}	$[NaHCO_3]_3$	*0.00005 mol*
pK_{a1}	$-\log_{10}K_{a1}$	*6.35*
pK_{a2}	$-\log_{10}K_{a2}$	*10.33*
pK_w	$-\log_{10}K_w$	*14*

The pH can be determined from W_a and W_b, using an implicit equation:

$$W_a=\left[H^+\right]-\frac{K_w}{\left[H^+\right]}-W_b\frac{\frac{K_{a1}}{\left[H^+\right]}+\frac{2K_{a1}K_{a2}}{\left[H^+\right]^2}}{1+\frac{K_{a1}}{\left[H^+\right]}+\frac{K_{a1}K_{a2}}{\left[H^+\right]^2}} \tag{5}$$

Solving the equation for $\left[H^+\right]$, the pH can be computed from

$$pH=\log_{10}\left[H^+\right] \tag{6}$$

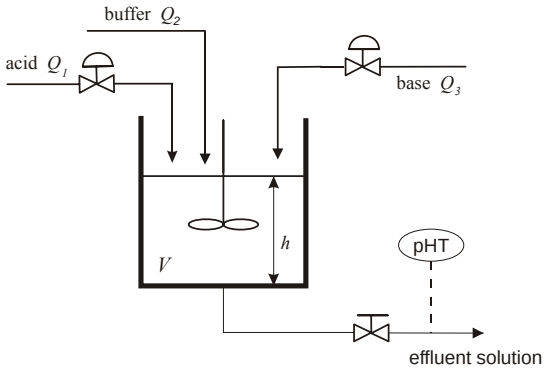

Figure 3. pH neutralization plant scheme.

4 Modeling the MIMO process

The steady state analysis (Figure 4) shows the nonlinearity of the process. Five operating areas with almost linear behavior can be identified. A local model network that describes the nonlinear plant was constructed using local auto-regressive with exogenous input (ARX) models of the first order. To obtain models relating to the *pH*, the base and acid flow rate was perturbed about their nominal values. Due to the relationship of the pH value to the acid and base flow-rate only the pH value at the time instant *k-1* is used for scheduling the local models. The local models had the form of a first-order ARX model:

$$\begin{aligned}&\left(I+\begin{bmatrix}a_1&0\\0&a_2\end{bmatrix}z^{-1}\right)\begin{pmatrix}y_1(k)\\y_2(k)\end{pmatrix}=\\&=\begin{bmatrix}b_{11}&b_{12}\\b_{21}&b_{22}\end{bmatrix}u(k-1)\end{aligned} \tag{7}$$

This open loop data was used to construct 5 local models at the operating point at the centers of the linear parts.

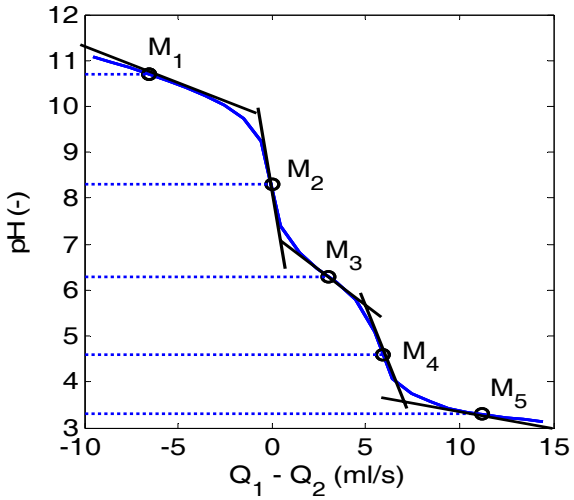

Figure 4. Steady state graph

To quantify the similarity between two systems a gap metric [9] is used. The gap metric is much more suitable to measure the distance between two linear systems than a metric based on norms. The gap metric between the local models associated with the clusters are computed. The gap metric for two SISO dynamic models is defined as:

$$\delta(M_1, M_2) = \sup_{\omega} \frac{|M_1(j\omega) - M_2(j\omega)|}{\sqrt{1+|M_1(j\omega)|^2}\sqrt{1+|M_2(j\omega)|^2}} \quad (8)$$

where $0 \leq \delta \leq 1$, $M_1(j\omega)$ and $M_2(j\omega)$ represent the frequency responses of the system M_1 and M_2 respectively. Two models have similar behavior in close-loop if the value of δ is close to 0 and behave differently for value of δ close to 1.

For MIMO system the value of gap metric can be deduced by comparing corresponding transfer functions

$$\delta(M_1, M_2) = \sup_{i,j}(\delta(G^1_{ij,}, G^2_{ij}))$$
$$M_1 = \begin{bmatrix} G^1_{11} & G^1_{12} \\ G^1_{21} & G^1_{22} \end{bmatrix} M_2 = \begin{bmatrix} G^2_{11} & G^2_{12} \\ G^2_{21} & G^2_{22} \end{bmatrix} \quad (9)$$

Table 2 Distances between the linear models measured using the gap metric

	1	2	3	4	5
1	0	0.72	0.03	0.24	0.01
2	0.72	0	0.70	0.53	0.72
3	0.03	0.70	0	0.21	0.03
4	0.24	0.53	0.21	0	0.24
5	0.01	0.72	0.03	0.24	0

Table 2 shows the gap metric between the pairs of five linear models representing the whole operating range of the nonlinear process. The obtained values shows the similarity between the models 1,3 and 5. The similarity between models 1, 3, and 5 can be explained physically by the fact that these models represent low-sensitivity regions.

The prediction of the local model network is given by:

$$\hat{y}(k+1) = \sum_{i=1}^{M} \mu_i(c_i, \sigma_i, \psi(t)) f_i(A, B, u(k), y(k)) \quad (10)$$

The centers of the validity functions were obtained from steady-state characteristic and parameters of local models by least-squares method using the data in the vicinity of the centre of the operating region. The remaining unknown parameter σ_i^2 from (6) is obtained by minimization the following criterion using the validation data:

$$MSE = \frac{1}{N-1} \sum_{k=1}^{N-1} \left(\hat{y}(k+1) - y(k+1)\right)^2 \quad (11)$$

The resulting distribution of the local models in the operating space of the system is shown in Figure 5.

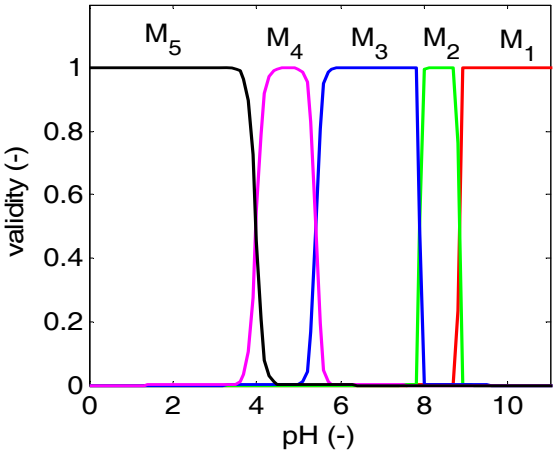

Figure 5. Validity functions of linear models

The modeling performance of the global model with 5 linear local models is depicted in Figure 6 where the output of the model is compared with output of the system. The obtained global model of the system is used for prediction within the framework of the predictive control. At each sampling point the global model was linearized using the scheduling vector and validity functions:

$$A = \sum_{i=1}^{M} \rho_i(c_i, \sigma_i, \psi(k)) A_i$$
$$B = \sum_{i=1}^{M} \rho_i(c_i, \sigma_i, \psi(k)) B_i \quad (12)$$

ISBN: 978-1-61804-004-6

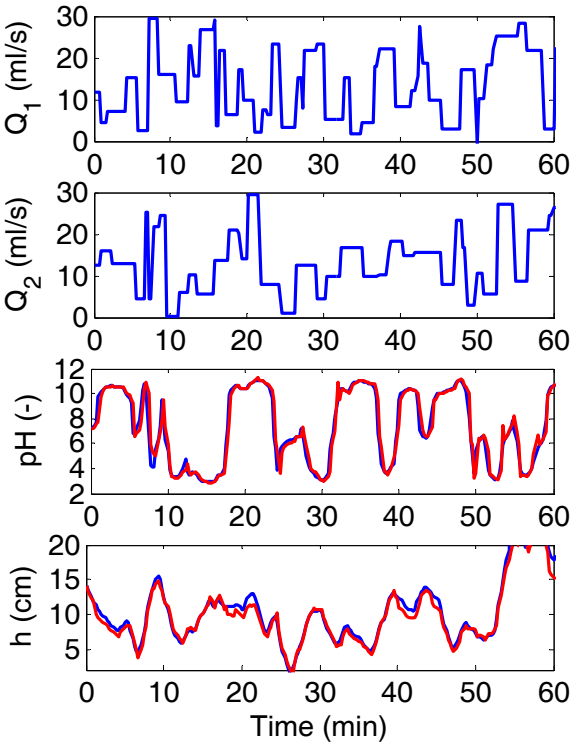

Figure 6. Modeling performance of LMN with 5 models

5 GPC control

A multivariable controlled auto-regressive integrated moving average (CARIMA) model for a system with R outputs and S inputs can be described by

$$\boldsymbol{A}(z^{-1})\boldsymbol{y}(k)=\boldsymbol{B}(z^{-1})\boldsymbol{u}(k-1)+\frac{\xi(k)}{\Delta} \tag{13}$$

where $\Delta=1-z^{-1}$, z^{-1} is the difference operator, whose function is to guarantee integral action in the controller to eliminate any offset. The terms $\boldsymbol{y}(k),\boldsymbol{u}(k),\xi(k)$ are the output, input, and noise vectors, respectively. The terms $\boldsymbol{A}(z^{-1}),\boldsymbol{B}(z^{-1})$ are the matrix polynomials of z^{-1} ,

$$\begin{aligned}\boldsymbol{A}&=\boldsymbol{I}+\boldsymbol{A}_1z^{-1}+\ldots+\boldsymbol{A}_{na}z^{-na}\\ \boldsymbol{B}&=\boldsymbol{B}_0+\boldsymbol{B}_1z^{-1}+\ldots+\boldsymbol{B}_{nb}z^{-nb}\end{aligned} \tag{14}$$

where *na* and *nb* are the orders of the model output and input, respectively. To design a GPC controller it is necessary to derive predictions *k*-step ahead:

$$\begin{aligned}\boldsymbol{Y}(t+k)&=\boldsymbol{G}\Delta\boldsymbol{U}(t)+\boldsymbol{S}(t+k)\\ \boldsymbol{S}(t+k)&=\boldsymbol{Y}_{k-1}\boldsymbol{Y}(t)+\boldsymbol{U}_{k-1}\Delta\boldsymbol{u}(t-1)\end{aligned} \tag{15}$$

where $\boldsymbol{S}$ represents the free response of the system and

$$\begin{aligned}\boldsymbol{Y}(t+k)&=\left[\boldsymbol{y}(t+1),\boldsymbol{y}(t+2),\ldots,\boldsymbol{y}(t+k)\right]^T\\ \boldsymbol{y}(t+1)&=[y_1(t+1),\ldots,y_R(t+1)]^T\\ \Delta\boldsymbol{U}(t)&=[\Delta\boldsymbol{u}(t),\ldots,\Delta\boldsymbol{u}(t+k-1)]^T\\ \Delta\boldsymbol{u}(t)&=\left[\Delta u_1(t),\ldots,\Delta u_S(t)\right]\\ \boldsymbol{Y}(t)&=[\boldsymbol{y}(t),\ldots,\boldsymbol{y}(t-na)]^T\\ \boldsymbol{y}(t)&=[y_1(t),\ldots y_R(t)]^T\end{aligned} \tag{16}$$

The cost function used in the GPC algorithms is defined as:

$$J=\sum_{k=1}^{N}\left\|\boldsymbol{Y}(t+k|t)-\boldsymbol{W}(t+k)\right\|_{\boldsymbol{Q}}^2+\sum_{k=1}^{N_u}\left\|\Delta\boldsymbol{U}(t+k-1)\right\|_{\boldsymbol{R}}^2 \tag{17}$$

where $\boldsymbol{W}(t+k)$ is the reference trajectory at a future time point k, N is the output prediction horizon, and N_u is the control increment horizon. This criterion can be rewritten in a matrix form

$$J=\frac{1}{2}\Delta\boldsymbol{u}^T\boldsymbol{H}\Delta\boldsymbol{u}+\boldsymbol{b}^T\Delta\boldsymbol{u}+\boldsymbol{f}_0 \tag{18}$$

where $\boldsymbol{H},\boldsymbol{b},\boldsymbol{f}_0$ are defined as

$$\begin{aligned}\boldsymbol{H}&=2(\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}+R)\\ \boldsymbol{b}&=2(\boldsymbol{S}-\boldsymbol{W})^T\boldsymbol{Q}\boldsymbol{G}\\ f_0&=(\boldsymbol{S}-\boldsymbol{W})^T(\boldsymbol{S}-\boldsymbol{W})\end{aligned} \tag{19}$$

Since the vector f_0 is a constant vector and does not have an effect on the quadratic programming result, the constrained optimization problem can be defined as

$$\begin{aligned}J&=\frac{1}{2}\Delta\boldsymbol{u}^T\boldsymbol{H}\Delta\boldsymbol{u}+\boldsymbol{b}^T\Delta\boldsymbol{u}\\ \boldsymbol{\Lambda}\Delta\boldsymbol{u}&\le\omega\end{aligned} \tag{20}$$

where $\boldsymbol{\Lambda}\Delta\boldsymbol{u}\le\omega$ defines the constraints for a control action increments.

Saturation constraints in the manipulated variables are imposed to take into account the minimum/maximum aperture of the valve regulating the base flow rate. A lower limit of 0ml/s and an upper limit of 20 ml/s are chosen for this variable. The prediction horizon was set to 8 samples as a result of using different values and comparing control performances. A control horizon of 4 samples was selected since further increase did not add significant improvement in terms of performance. The weighting matrix $\boldsymbol{Q}$ associated

ISBN: 978-1-61804-004-6

with the error from set point was set two times greater than matrix $\boldsymbol{R}$ associated control signal changes.

$$\boldsymbol{Q} = 2I, \boldsymbol{R} = I \tag{21}$$

The model predictive control algorithm described in Section 5 was implemented using the "quadprog" function in MATLAB's Optimization Toolbox to minimize the cost function. To reduce the on-line computational load, the control sequence computed at the step $k - 1$ was shifted backwards and used as an initial guess for the computation of the future controller output at time k. The resulting control courses for stepwise set-point changes are shown in Figure 7.

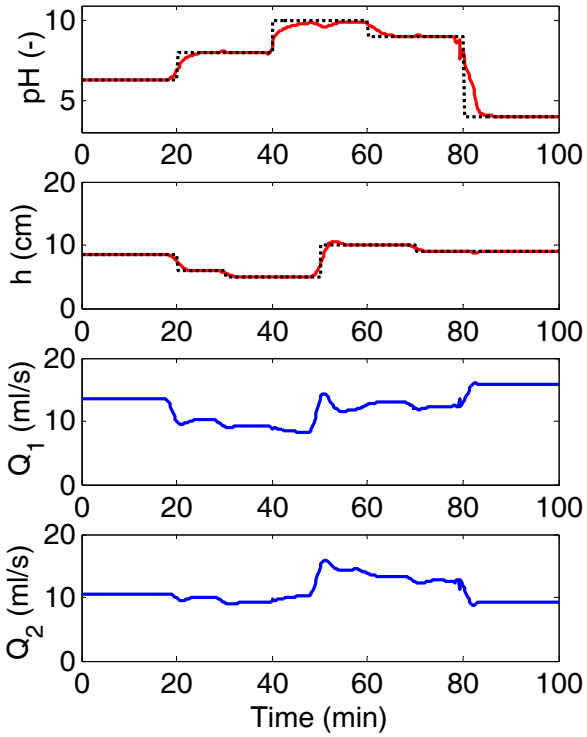

Figure 7. The LMN-GPC performance when subjected to step-like set point changes

6 Conclusion

A nonlinear model-based predictive control strategy based on a local model network has been presented. The simplicity of the modeling and the performance of the controller make the design very attractive for a designer. The use of a local model network makes the global nonlinear model more transparent and user-friendly than a model based on a neural network, where the information given by the parameters is not very clear to the user. A linearization of the LMN was used as a system model for the GPC algorithm. The advantage of this technique is that time-consuming numerical optimization methods and an uncertainty in the convergence to the global optimum, which are typically seen in conventional nonlinear model-based predictive control, are avoided. Moreover, the control actions obtained based on local incremental models contain integration actions that can naturally eliminate static offsets. This approach was implemented experimentally to control the pH and level in pH neutralization process.

Acknowledgement

The work has been supported by Czech Grant Agency under the grants 102/09/P243.

References:

[1] E.F. Camacho, C. Bordons, *Model Predictive Control,* Springer, London, 1999.

[2] J.M. Maciejowski, *Predictive Control with Constraints*, Prentice Hall, Harlow, 2002.

[3] S.J. Qin, T.A. Badgwell, An overview of industrial model predictive control technology, *Chemical Process Control,* 1996, pp. 232-256.

[4] P. Saha, S.H. Krishnan, V.S.R. Rao, S.C. Patwardhan, Modeling and predictive control of MIMO nonlinear systems using Wiener-Laguerre models, *Chemical Engineering Communications*, Vol. 191, No. 8, 2004, pp. 1083-1119.

[5] R. Murray-Smith, T.A. Johansen, *Multiple Model Approaches to Modelling and Control,* Taylor and Francis, London, 1997.

[6] O. Galan, J.A. Romagnoli, A. Palazoglu, Real-time implementation of multi-linear model-based control strategies—an application to a bench-scale pH neutralization reactor, *Journal of Process Control*, Vol. 15, No. 5, 2004, pp. 571–579.

[7] Z.K. Xue, S.Y. Li, Multi-model modelling and predictive control based on local model networks, *Control and Intelligent Systems archive*, Vol. 34, No. 2, 2006, pp. 105 - 112.

[8] T.A. Johansen, B.A. Foss, Identification of nonlinear system structure and parameters using regime decomposition, *Automatica*, Vol. 31, No. 2, pp. 321-326.

[9] T.T Georgiou, M.C. Smith, Optimal robustness in the gap metric. *IEEE Transactions on Automatic Control*, Vol. 35, 1990, pp. 673-686.