15th Triennial World Congress, Barcelona, Spain

A NEURO-FUZZY MODEL PREDICTIVE CONTROLLER APPLIED TO A PH-NEUTRALIZATION PROCESS

Jonas B. Waller and Hannu T. Toivonen

Department of Chemical Engineering, Åbo Akademi University, FI-20500 Åbo, Finland

Abstract: This paper addresses the issue of controlling nonlinear processes by the use of the nonlinear model predictive control formulation. To handle the nonlinearities, a neuro-fuzzy process model is suggested as a means to model processes with strong nonlinearities depending on the operating region. In this paper the neuro-fuzzy approach is used for the modelling and control of a strongly nonlinear pH neutralization process, both in the face of set-point changes and in the face of unmodelled disturbances.

Keywords: Process control, pH control, nonlinear control, neural network models, nonlinear models.

1. INTRODUCTION

Model predictive control (MPC) has received a strong position when it comes to industrially implemented advanced control methodologies, especially in the refinery and petrochemical fields, see for example Qin and Badgwell (1998) and Takatsu *et al.* (1998). The MPC formulation is commonly formulated for linear models. In many problems relevant in the process control field today, however, the plant under control shows a strongly nonlinear behavior. Nonlinear MPC (NMPC), simply put, is model predictive control, where a nonlinear process model is used for prediction purposes, as opposed to a linear model used for prediction purposes in the case of linear MPC (Rawlings *et al.*, 1994; Henson, 1998; Camacho and Bordons, 1999; Rawlings, 2000).

One main difficulty with nonlinear MPC is that the nonlinear models needed often are complex and give rise to computationally burdensome optimization problems. Since MPC requires the solution of an optimization problem on-line at every sampling instant, the computational simplicity is crucial. Furthermore, process models used in nonlinear MPC control strategies are often difficult to interpret and understand, thus making the choice of tuning-parameters a quite ad hoc procedure (Henson, 1998; Rawlings, 2000).

A neuro-fuzzy modelling technique, recently presented in the literature (Hu *et al.*, 1998; Hu *et al.*, 1999), is here used for NMPC of a pH neutralization process. The controller has been introduced in Waller *et al.* (2000). The structure of the neuro-fuzzy model is physically motivated through linear input/output modelling techniques, as the model consists of a network of a global linear predictor and several local linear predictors. Similar modelling techniques have been introduced in e.g. Johansen and Foss (1993). The simplicity of the model can be argued to contribute to making the procedure of tuning the NMPC system more transparent when using the neuro-fuzzy predictor.

2. NONLINEAR MODEL PREDICTIVE CONTROL

Nonlinear model predictive control is an open-loop optimal control sequence calculation where the main characteristics of the controller are (Camacho and Bordons, 1999)

(1) Explicit use of a nonlinear model to predict the process output at future time instants (a horizon).
(2) Calculation of a control sequence minimizing an open-loop objective function.
(3) Receding strategy so that at each instant the horizon is displaced towards the future, which involves the application of the first control signal of the sequence calculated at each step.

We can briefly formulate the problem for nonlinear MPC as follows. The process is assumed to have the form of a nonlinear discrete-time input-output mapping, i.e.,

$$y(k) = g(\varphi(k)) + e(k) \tag{1}$$

$$\varphi(k) = [y(k-1), ..., y(k-n), u(k-d-1), \\ ..., u(k-m-d)]^T \tag{2}$$

Where y denotes the process output, u the process input, n is the number of old outputs in the model, m the number of old inputs and d is the delay. The nonlinear mapping is denoted with g and e denotes an error term. The sampling index is denoted k.

The open-loop optimization problem which is solved at every sampling instant can be formulated as minimizing a loss function comprising of future errors between set-point values and predicted process outputs and the future control moves (or the change in them). The loss function might also include a terminal constraint, or a terminal penalty term. The minimization is performed with respect to the future control moves, taking possible constraints into account. A good treatment of relevant optimization methods is offered in e.g. Gopal and Biegler (1998).

We use the change of the control input $\Delta u(k)$ in the cost function, in order to eliminate steady state offsets and also to emphasize the smoothness of the control input. In process control, the smoothness of the control action is often of significance, in order to e.g. reduce wear on actuators. We will thus minimize a cost function given by

$$J = \sum_{j=N_1}^{N_y} \sigma(j)[y_r(k+j) - \hat{y}(k+j|k)]^2 \\ + \sum_{j=1}^{N_u} \lambda(j)[\Delta u(k+j-1)]^2 \tag{3}$$

The weights on the process output are denoted with $\sigma(j)$ and on the process input with $\lambda(j)$.

The capability to handle constraints is considered one of the main motivations to use (nonlinear) MPC. The optimization problem formulated in equation (3) is in the case of constraints present solved subject to the input inequality constraints

$$u_{\min} \leq u(k+j|k) \leq u_{max}, \; 0 \leq j \leq N_u - 1 \tag{4}$$
$$\Delta u_{\min} \leq \Delta u(k+j|k) \leq \Delta u_{max}, \; 0 \leq j \leq N_u - 1$$

and the output inequality constraints

$$y_{\min} \leq y(k+j|k) \leq y_{max}, \quad 1 \leq j \leq N_y \tag{5}$$

3. NEURO-FUZZY MODELLING

A key issue to address when designing a NMPC controller is the choice of process model, i.e. the choice of the structure of the (nonlinear) process model to be used. Here we can roughly distinguish between two (or three) kinds of models. Firstly there are models based on fundamental relationships and secondly there are models based on empirical data. The third kind of model would be one combining fundamental and empirical modelling techniques (which usually is referred to as a hybrid model).

For a presentation of a few of the nonlinear process models utilized for NMPC in the recent literature, see Waller *et al.* (2000).

Here, we shall consider a neuro-fuzzy (also referred to as a quasi-ARMAX) modelling scheme as a model to be used in the NMPC formulation, which has been introduced by Hu *et al.* (1998). Basically, the idea of the model is to express the process as comprising of two parts, a global linear model and a network of local linear models. The nonlinear model is then a combination of this global model and some of the local models, as activated according to a fuzzy logic system. The fuzzy logic system activates local models based on the information in the whole regression vector. Furthermore, we include an explicit evaluation of the modelling error between the actual process output and the output as given by the neuro-fuzzy model in the overall model. Starting from the process description in equation (1), the model can be approximated as follows. The nonlinear mapping is approximated using the neuro-fuzzy model according to

$$g(\varphi(k))) \approx g_0 + \varphi_{NL}^T(k)\Theta \tag{6}$$

where g_0 is a bias term and

$$\Theta = [\theta^T \; \omega_{11} \ldots \omega_{rL}]^T$$

$$\varphi_{NL}(k) = [\varphi^T(k) \quad \varphi^T(k) \otimes \varphi_{N_f}^T(k)]^T$$

$$\varphi_{N_f}^T(k) = [N_f(p_j, \varphi(k)), \quad j = 2, \ldots L]$$

Θ is the parameter vector, θ are the parameters of the global model and ω_{ij} are the parameters of the local models. Kronecker production is indicated by $\otimes$. The parameters Θ can be evaluated using e.g. a recursive least squares algorithm. φ is the regression vector and N_f is the network of fuzzy logic systems. p_j in turn are the position and scale parameters of the fuzzy logic system. See Hu *et al.* (1998) for details.

As long as the process is operated at its nominal state, and the error between the process and the neuro-fuzzy model is small, there is no obvious need for compensating for the plant/model mismatch. However, in the presence of e.g. disturbances, the mismatch might be significant and thus the modelling error should be compensated for explicitly. In this paper this is done through approximation of the modelling error $e(k)$ according to

$$\hat{e}(k|k) = \hat{e}(k-1|k-1) + \alpha(y(k) - (g_0 + \varphi_{NL}^T(k)\Theta) - \hat{e}(k-1|k-1)) \quad (7)$$

and assuming a constant error over the prediction horizon, i.e.

$$\hat{e}(k+j|k) = \hat{e}(k|k), \qquad j = 1, 2, ..., N_y \quad (8)$$

The forgetting factor α is chosen in order to give the error evaluation a memory. In our case, we chose $\alpha = 0.5$ as a compromise between smooth error evaluation and quick response to sudden disturbances. This provides the controller with integral action.

The predicted output is thus computed recursively according to

$$\hat{y}(k+1|k) = g_0 + \hat{\varphi}_{NL}^T(k+1|k)\Theta + \hat{e}(k+1|k)$$

based on the regression vector

$$\hat{\varphi}(k+1|k) = [y(k) \quad y(k-1) \; ... \quad u(k-d) \quad u(k-d-1) \; ...])$$

and the two-step ahead prediction

$$\hat{y}(k+2|k) = g_0 + \hat{\varphi}_{NL}^T(k+2|k)\Theta + \hat{e}(k+2|k)$$

in turn based on

$$\hat{\varphi}(k+2|k) = [\hat{y}(k+1|k) \quad y(k) \; ... \quad u(k-d+1) \quad u(k-d) \; ...])$$

and similarly for the rest of the prediction horizon.

4. CONTROL OF A PH-NEUTRALIZATION PROCESS

4.1 *The pH-neutralization Process*

The process used to illustrate the control results is a pH-neutralization process, previously studied and presented in the literature (Sandström and Gustafsson, 1994; Gustafsson *et al.*, 1995). The process consists of a continuous stirred tank reactor with a constant volume and feed flow. The feed stream is a water-phosphoric acid solution of varying concentration. The control stream is a concentrated solution of calcium hydroxide. Figure 1 shows a schematic illustration of the pH process. The objective of the controller is to control the pH value throughout the range of operating regions, and in the face of varying disturbances. This process has recently been used to illustrate multimodel robust control in Nyström *et al.* (1998), and in Nyström *et al.* (1999). See the references (Sandström and Gustafsson, 1994; Gustafsson *et al.*, 1995; Nyström *et al.*, 1998; Nyström *et al.*, 1999) for details on the process and the model used to simulate the real process here.

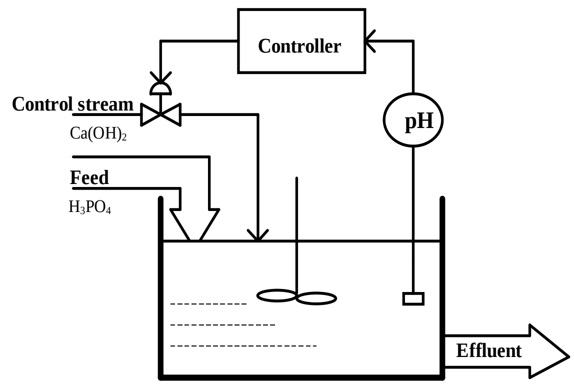

Fig. 1. A schematic illustration of the pH process.

In figure 2 the incremental gain of the process is shown as a function of the operating region, when the process is operating at its nominal feed flow rate and composition. The gain is numerically calculated from the titration-curve as $\frac{\partial y}{\partial u}$ and plotted as a function of the pH-value in the upper graph, and as a function of the process input u in the lower graph. The incremental gain of the process changes from around 170 at pH=2.5 to over 7600 at pH=4.55, down again to about 330 at pH=6. The nonlinearities are not merely static, also dynamic nonlinearities are present.

4.2 *Identification*

We identify a neuro-fuzzy model by running a sequence of small step changes through the entire operating region, gradually moving the sequence through the titration curve, both upwards and downwards. This is done in order to excite the process in all regions represented by the set of local models in the network. The feed flow rate and the feed composition are kept constant during the whole identification sequence. The sample time of the identification is 0.2 minutes. All time units unless stated otherwise are in minutes. The conver-

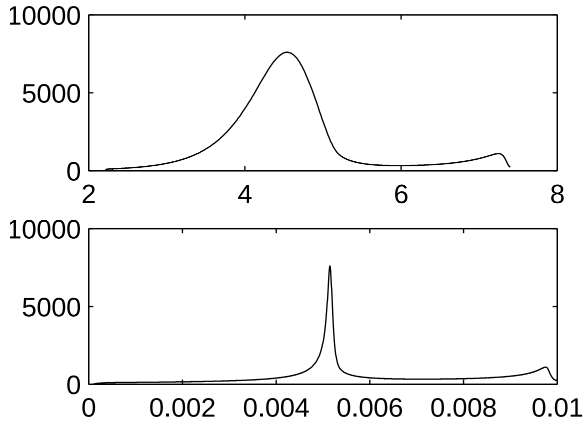

Fig. 2. The process gain as a function of the pH value (upper graph) and as a function of the process input (lower graph).

gence of the neuro-fuzzy model requires repeated calculations with the same data, in order to get a model with sufficient modelling capabilities.

The parameters for the neuro-fuzzy model are chosen such as to achieve a sufficient modelling accuracy with as few parameters as possible. This can be done due to the nature of the model where prior process knowledge can be utilized. We choose the model structure for the global model and for all local models as having $n = 2$, $m = 2$ and $d = 5$ (i.e. 1 minute time delay in the process). These parameters are sufficiently large to model the process, but small enough to keep the numerical calculations at a reasonable level. Numerical tests have not shown any significant improvement in the prediction ability when the number of parameters are increased. The activation regions of the local models are linearly spaced in terms of the pH value within the regions of interest, for the pH-value between 2.2 and 7.4, and corresponding values for the process input between 0 and 0.01. In the model, 30 local models are used. Further, we scale the process input by the factor of 500, so that the magnitude of the process input and output are the same.

Better prediction in the high gain region can be achieved with a larger number of local models, at the cost of slower computations. However, for the purpose of this study, the prediction accuracy with 30 local models is sufficient.

4.3 *Control Results*

The NMPC controller minimizes a loss function at every sampling instant. A number of future control moves (the control horizon) is calculated each time, and the first control move of this control horizon is implemented. The calculations are then repeated at the next sampling instant.

Although the MPC formulation is considered to have an intuitive appeal, the choice of all relevant parameters for the MPC controller can, regardless, be quite cumbersome (Rawlings, 2000; Henson, 1998; Rohani *et al.*, 1999). This is the case especially when using nonlinear models. A significant amount of heuristics might in that case be necessary.

The choice of the prediction and the control horizons is first performed. Based on the recommendations in (Rohani *et al.*, 1999), we chose a control horizon as small as possible, in order to reduce the computational burden. In our example, a control horizon of $N_u = 2$ is sufficient. The prediction horizon is chosen as $N_1 = 6$ (the first process output affected by the calculated control moves) and $N_y = 12$. The end of the prediction horizon should be large enough so that the control moves have an effect on them. Thus, $N_y \geq N_u + d + 1$ should hold. If the prediction horizon is longer than the control horizon, the future control moves beyond the control horizon are assumed constant in calculation of the prediction horizon. We keep the weight on the process output (the pH value) constant, $\sigma(j) = 1$, and only adjust the weight on the inputs ($\lambda(j)$).

The constraints formulated for this problem are hard constraints on the absolute value of the inputs,

$$u_{\min} \leq u(k+j) \leq u_{\max} \tag{9}$$

with $u_{\min} = 0$ and $u_{\max} = 0.01$.

A series of set-point changes is first studied. The composition of the feed is at its nominal value, i.e. the same composition as used during the identification, through-out the set-point simulations. With a constant weight, $\lambda(j) = \lambda_0$, on the input, the response in the low gain region will be too sluggish, if the response in the high gain region is appropriately quick. See the simulation results in figure 3 with $\lambda(j) = 50$.

Thus, the input weight must be relative to the operating region. In our case, we chose an input weight relative to the predicted process outputs, i.e.

$$\lambda(j) = \lambda_0 W_s(j + N_1), \qquad j = 1, 2, ..., N_u \tag{10}$$

$$W_s(k) = \frac{1}{max(K_P)} K_P(\hat{y}(k))$$

$K_P(\hat{y})$ is the incremental process gain as a function of the predicted process output, as presented in figure 2, the upper graph. Results with this relative input weighting ($\lambda_0 = 150$) is presented in figure 4. Further tuning can of course be performed, if considered necessary. In our example, the response to set point changes in the low gain regions can be speeded up by decreasing the input weight in the low gain regions. A slightly modified

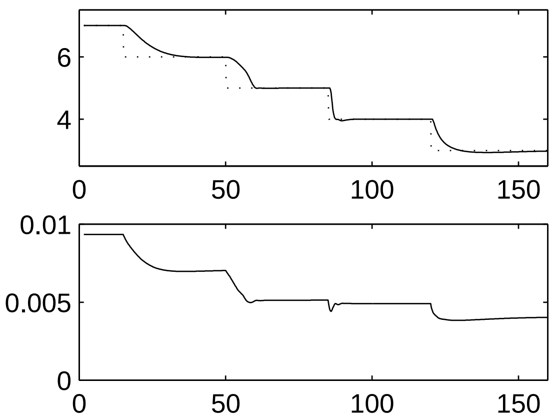

Fig. 3. pH value as resulting from a series of set-point changes. Upper curve shows the actual process output (solid line) and the set-points (dotted line), and the lower graph shows the control effort, i.e. the process input. Time in minutes on the x-axis. The input weight is constant over the entire operating range.

tuning based on the relative input weighting is presented in figure 5. The modifications simply consist of decreasing the input weights in the regions where the response according to figure 4 is too sluggish.

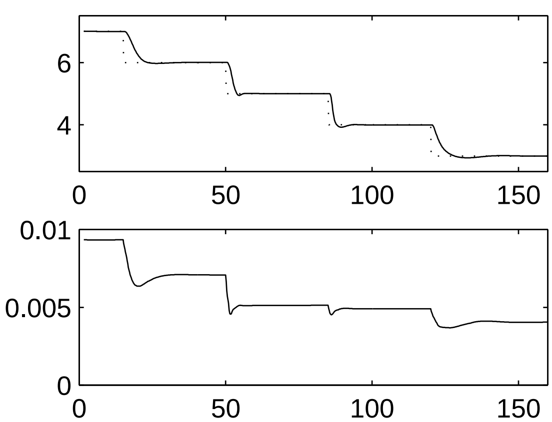

Fig. 4. pH value as resulting from a series of set-point changes. Upper curve shows the actual process output (solid line) and the set-points (dotted line), and the lower graph shows the control effort, i.e. the process input. Time in minutes on the x-axis. The input weight is scaled relative to the process gain.

If we look at the case with unmodelled disturbance in the feed flow concentration, the results are as presented in figure 6. The process is run at set-point values of pH = 3, 4, 5 and 6. The disturbance in the feed concentration occurs at time = 15 minutes from 0.01 moll^{-1} to 0.011 moll^{-1} and at time = 65 minutes back to 0.01 moll^{-1}. This is the same disturbance as studied in Nyström *et al.* (1998) and Nyström *et al.* (1999). It is worth mentioning that this disturbance has not been used in any way during the identification, design nor tuning of the controller.

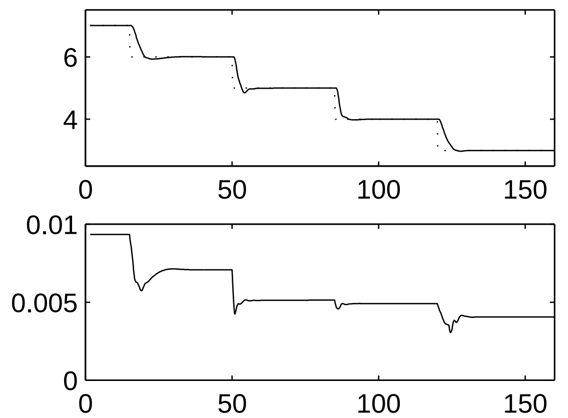

Fig. 5. pH value as resulting from a series of set-point changes. Upper curve shows the actual process output (solid line) and the set-points (dotted line), and the lower graph shows the control effort, i.e. the process input. Time in minutes on the x-axis. The input weight is modified, based on the relative weighting.

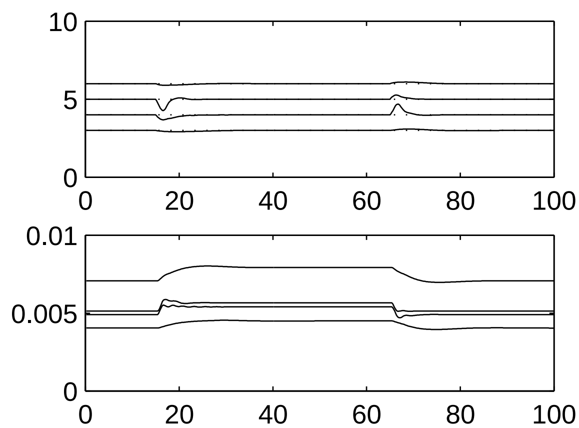

Fig. 6. pH value (upper graph) as resulting from a change in feed composition (a step disturbance), at four different set-point values, pH=3, 4, 5 and 6. Lower graph shows the corresponding process inputs. Time in minutes on the x-axis. The input weight is relative to the pH value.

5. DISCUSSION AND CONCLUSIONS

A neuro-fuzzy approach to process control using the nonlinear model predictive control formulation has been presented. The model used for predicting the future outputs consists of a network of a global linear predictor and several local linear predictors, and the interpolation between the models is done using a fuzzy logic system. This modelling combines high prediction capabilities with the fundamental understanding and simplicity of linear ARMAX modelling. Furthermore, as the model has a clear structure adjusted for the problem at hand, the amount of parameters is kept low. The neuro-fuzzy model is capable of modelling the behavior of a highly nonlinear pH neutralization process to a high accuracy.

As the model is designed to model the process accurately at different operating regions, the results with excellent control quality regardless of operating region are not surprising. We note, however, that since the MPC formulation expects relative weights on input changes and output errors, this relation must be adjusted depending on the particular operating region. This is in the example in this paper done by scaling the input weight with the incremental gain of the process.

The NMPC controller works very well on a highly nonlinear pH-neutralization process, and combines ease of use with good control quality. The control quality with respect to eliminating the effect of disturbances in the feed flow concentration is of the same quality as the best results achieved in Nyström *et al.* (1999), and the results achieved here are more successful then the results in Nyström *et al.* (1998) when it comes to set-point changes.

The controller would benefit from further studies. This paper acknowledges the problem of tuning NMPC controllers, and the vast amount of heuristics needed, especially in the choice of parameters for the used nonlinear model, control and prediction horizons and the choice of weights for the inputs and the outputs. Although in this paper the choice of parameters for the identified model is simplified due to the simplicity of the model structure, this weakness of the NMPC formulation in general should be addressed.

ACKNOWLEDGMENTS

Jonas Waller gratefully acknowledges financial support from Svenska Tekniska Vetenskapsakademien i Finland. The work was also supported by the Academy of Finland under grant 44031.

6. REFERENCES

Camacho, E.F. and C. Bordons (1999). *Model Predictive Control*. Springer Verlag.

Gopal, Vipin and Lorenz Biegler (1998). Large scale inequality constrained optimisation and control. *IEEE Control Systems*.

Gustafsson, Tore K., Bengt O. Skrifvars, Katarina V. Sandström and Kurt V. Waller (1995). Modeling of pH for control. *Ind. Eng. Chem. Res.* **34**(3), 820–827.

Henson, M.A. (1998). Non-linear model predictive control: Current status and future directions. *Computers chem. Engng.* **23**, 187–202.

Hu, J., K. Hirasawa and K. Kumamuru (1999). Adaptive predictor for control of nonlinear systems based on neurofuzzy models. In: *Proc. ECC*. Karlsruhe, Germany.

Hu, J., K. Kumamuru, K. Inoue and K. Hirasawa (1998). A hybrid quasi-ARMAX modeling scheme for identification of nonlinear systems. *Trans. SICE* **34**, 977–986.

Johansen, Tor A. and Bjarne A. Foss (1993). Constructing NARMAX models using ARMAX models. *Int. J. Control* **58**(5), 1125–1153.

Nyström, Rasmus H., Katarina V. Sandström, Tore K. Gustafsson and Hannu T. Toivonen (1998). Multimodel robust control applied to a pH neutralization process. *Computers chem. Engng* **22**, S467–S474.

Nyström, Rasmus H., Katarina V. Sandström, Tore K. Gustafsson and Hannu T. Toivonen (1999). Multimodel robust control of nonlinear plants: a case study. *Journal of Process Control* **9**, 135–150.

Qin, S. Joe and T.A. Badgwell (1998). An overview of non-linear model predictive control applications. In: *Proceedings Workshop on NMPC*. Ascona, Switzerland.

Rawlings, J.B. (2000). Tutorial overview of model predictive control. *IEEE Control Systems* **20**(3), 38–52.

Rawlings, J.B., E.S. Meadows and K.R. Muske (1994). Non-linear model predictive control: A tutorial and a survey. In: *Preprints IFAC Symposium ADCHEM '94*. Kyoto Research Park, Kyoto, Japan.

Rohani, S., M. Haeri and H.C. Wood (1999). Modeling and control of a continuous crystallization process. part 2. model predictive control. *Computers chem. Engng.* **23**, 279–286.

Sandström, K.V. and T.K. Gustafsson (1994). A study of the dynamics of calcium-phosphate precipitation in pH-control systems. Technical report. Process Control Laboratory, Åbo Akademi University.

Takatsu, H., T. Itoh and M. Araki (1998). Future needs for the control theory in industries - report and topics of the control technology survey in japanese industry. *J. Proc. Cont.* **8**(5-6), 369–374.

Waller, Jonas B., Jinglu Hu and Kotaro Hirasawa (2000). Nonlinear model predictive control utilizing a neuro-fuzzy predictor. In: *Proceedings of the IEEE Conference on SMC 2000*. Nashville, Tennessee, USA.