Proc. IASTED - ASC'2000, July, Banff, Canada, 2000

EXTENDED NEURAL MODEL PREDICTIVE CONTROL OF NON-LINEAR SYSTEMS

PAULO GIL[†‡] JORGE HENRIQUES[‡] ANTÓNIO DOURADO[‡] H. DUARTE-RAMOS[†]

[†] CISUC- Informatics Engineering Department, UC
Pólo II, Pinhal de Marrocos, 3030 Coimbra
Phone: +351 239790000, Fax: +351 239701266
Portugal

[‡] Electrical Engineering Department, UNL
2825-114 Monte de Caparica - Caparica
Phone: +351 212948545, Fax: +351 212948532
Portugal

ABSTRACT

A neural model-based predictive control scheme is proposed for dealing with steady-state offsets found in standard MPC schemes. This structure is based on a constrained local instantaneous linear model-based predictive control methodologies together with a static offset pre-filter for assuring free tracking errors and disturbance rejection features. A non-linear state-space neural network architecture trained offline is used for modelling purposes, from which linear models are extracted by Taylor series expansion at each sampling time. Results from experiments show that this extended MPC scheme ensures good tracking and disturbance rejection performances.

Keywords: Non-linear systems; recurrent neural networks; model predictive control; constrained optimisation.

1 INTRODUCTION

In recent years, there has been a growing interest in developing and applying model-based predictive control methodologies (MPC) in solving practical control problems, mostly in the processing industries. The success of MPC techniques can be attributed mainly to its inherent abilities to handle either multivariable input or multivariable output constraints [1], while requiring that the resulting control sequence is, in some way, optimal with respect to a prespecified cost function.

In the kernel of all MPC methodologies lies the moving horizon approach proposed in the early 1960’s by Propoi, 1963 [2]. However, it was not until the late 1970’s with the works of Richalet *et al*. [3] and Cutler and Ramaker [4] that successful implementations of model predictive control schemes in the processing industry were reported, making use of a linear model description of the underlying plants. Given the non-linear nature of most industrial processes, some research efforts have been placed in the last few years in building non-linear model predictive control (NMPC) schemes, as testified by the considerable number of papers found in the literature (see *e.g.* [5], [6] and [7]).

As in many non-linear systems it is not an easy task to come up with an accurate enough physical model of the plant, which is required by MPC techniques, one may turn to black-box models to describe the system’s non-linear dynamics. Among available structures [8], neural networks have proved to work quite well in the identification of non-linear systems on the basis of input-output data [9]. Despite neural networks are well known universal approximators [10], [11], they are quite dependent on the quality of the data set. This feature together with a bounded number of iterations, within the training phase, leads inexorably to a model mismatch, which in turn is responsible for a static error or, in worst case, giving rise to the instability of the feedback system.

In this paper we address the design of an extended constrained local instantaneous linear model predictive control (LIMPC) scheme based on state-space neural networks models, assuring that control errors are driven to zero in a finite time and, at same time, exhibiting good disturbance rejection performance. Within this scope, to prevent from static offsets a pre-filter is incorporated in the control loop in such a way that, particularly, in the vicinity of the set-points the reference signals fed to the MPC structure are previously changed on basis of current deviations. Since the control error is weighted by a decreasing exponential, as a function of the control error itself, only small errors are actually taken into consideration within the static offset compensator and thus undesirable windup effects are avoided.

2 RECURRENT NEURAL NETWORKS MODELLING

In the wake of the recovered interest in the field of artificial neural networks, following a period of some apathy during 1970's, Narendra and Parthasarathy in a seminal paper [12] have proposed to use neural networks in the context of identification and control of non-linear systems. Since then, and particularly in the last few years, neural networks have become a reliable conceptual tool for describing non-linear systems dynamics given input-output measurements collected from the plant.

Neural networks essentially consists of a number of neurons, known also as processing element or cell unit, which are connected to each other by means of a linear or non-linear activation function. Weights playing a role analogous to the density of neurotransmitters in biological synapses being adjusted in the learning stage quantifies the synaptic strength between neurons.

Topologically neural networks can be arranged in a feedforward or recurrent way. In a feedforward network synaptic signals flow via unidirectional connections between consecutive layers, while in the recurrent case it is allowed feedback loops in a number of cell units. For a comprehensive review on neural networks the reader is referred to [13], [14].

Consider that a description of its dynamics is required for the general non-linear discrete-time system given in (1) and that is achieved by finding an appropriate non-linear mapping on the basis of neural networks.

$$\begin{aligned} x(k+1) &= f\left(x(k), u(k)\right) \\ y(k) &= g\left(x(k), u(k)\right) \end{aligned} \qquad (1)$$

where $f : \mathrm{R}^n \times \mathrm{R}^m \to \mathrm{R}^n$ and $g : \mathrm{R}^n \times \mathrm{R}^m \to \mathrm{R}^p$ are appropriate non-linear functions; $x(k) \in \mathrm{R}^n$ is the current state vector; $u(k) \in \mathrm{R}^m$ is the current control vector; $y(k) \in \mathrm{R}^p$ denotes the current output vector.

For this purposes, we use in this paper the recurrent neural network architecture depicted schematically in Figure 1. The choice for this class of neural networks was made in straight connection to the fact that the incorporation of feedback enables to describe the plant dynamics and, moreover, that state-space neural predictors are not only less demanding with respect to the number of parameters than input-output neural networks [15] but they are also quite suitable for using within the MPC framework.

As such, the problem of finding an appropriate model of a given system is here converted into looking for a convenient number of neurons for each layer and subsequently in learning a non-linear mapping, which is established in a supervised way, using input and output data.

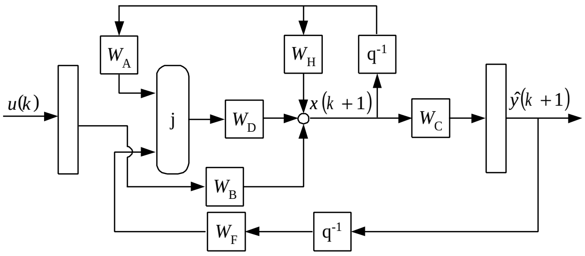


Figure 1: Recurrent neural network architecture.

In this architecture $x(k) \in \mathrm{R}^n$ is denoted the current neural state vector, $\hat{y}(k) \in \mathrm{R}^p$ is the predicted output vector, W_A, W_B, W_C, W_D, W_F and W_H are weight matrices of appropriate dimensions, q^{-1} is the backward shift operator and φ is the hyperbolic tangent activation function. Its dynamics is described in the state-space form as follows:

$$\begin{aligned} x(k+1) &= W_D \tanh\left[\left(W_A + W_F W_C\right) x(k)\right] + W_H\, x(k) \\ &\quad + W_B\, u(k) \\ \hat{y}(k) &= W_C\, x(k) \end{aligned} \qquad (2)$$

The neural network above is trained offline in the conventional supervised way by minimizing a sum of squared prediction errors (3) with respect to the network's weights using the Levenberg-Marquardt algorithm.

$$J(W) = \sum_{k=1}^{N} \left[y(k) - \hat{y}(k|W)\right]^2 \qquad (3)$$

According to the Levenberg-Marquardt method weights are updated iteratively as follows:

$$W^{i+1} = W^i - \left[J^T\left(W^i\right) J\left(W^i\right) + l^i\, \mathrm{I}\right]^{-1} \nabla J\left(W^i\right) \qquad (4)$$

where $J(W)$ is the cost-function, $\nabla J(W)$ denotes its gradient, $l \in \mathrm{R}^+$ and I is an identity matrix of appropriate dimensions.

This algorithm has the very attractive feature that as l is increased it moves towards the steepest descent method with the learning rate given as $1/l$, while decreasing l the algorithm becomes Gauss-Newton. Therefore, the algorithm provides a quite interesting compromise between the speed of Newton's method and the guaranteed convergence of steepest descent algorithm [16]. A straightforward strategy to select and update l can be found in [17].

3 EXTENDED LIMPC FORMULATION

Model-based predictive control is a discrete-time technique for which an explicit dynamic model of the plant is used to predict the system's outputs over the finite prediction horizon P when control actions are manipulated over the finite control horizon M .

At time step k , the optimiser computes on-line the optimal open-loop control actions sequence in such a way that the predicted outputs follows a pre-specified reference while taking into account possible hard and soft constraints. From the computed sequence, only the control action $u(k \mid k)$ is actually implemented on the plant over the time interval $[k, k+1)$ (Figure 2). Next, the prediction and control horizons are shifted ahead by one step and a new optimisation problem is solved by taking into consideration the most recent measurements from the plant, and the control actions fed to the plant in the previous time step, $u(k \mid k)$.

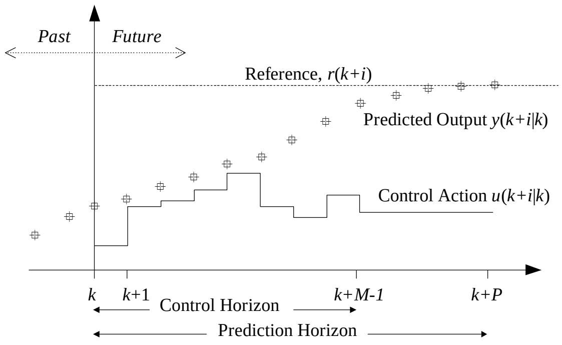


Figure 2: Receding horizon implementation of MPC.

Let the first order Taylor expansion of a general non-linear system be described in the discrete-time state-space form as follows:

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Gamma u(k) + h \\ y(k) &= \Xi x(k) \end{aligned} \tag{5}$$

where $\Phi \in \mathrm{R}^{n \times n}$, $\Gamma \in \Re^{n \times m}$ and $\Xi \in \mathrm{R}^{p \times n}$ are the state, input and output matrices, respectively; $x(k) \in \mathrm{R}^{n}$ is the state vector, $u(k) \in \mathrm{R}^{m}$ is the control vector and $y(k) \in \mathrm{R}^{p}$ the output vector, as mentioned above; $h \in \mathrm{R}^{n}$ is a constant vector related to the first term of the Taylor series.

Assuming a 2-norm for the cost functional and taking into consideration linear constraints on the inputs and outputs of the system together with bounds on the rate of change of control actions, the open-loop optimisation problem can be stated as follows:

$$\begin{aligned} \min_{u} J = \min_{u} \Bigg\{ & \sum_{i=1}^{P} \| y(k+i \mid k) - r(k+i) \|_{Q_i}^{2} \\ & + \sum_{i=0}^{P-1} \| u(k+i \mid k) \|_{R_i}^{2} + \sum_{i=0}^{M-1} \| \Delta u(k+i \mid k) \|_{S_i}^{2} \Bigg\} \end{aligned} \tag{6}$$

subject to the system dynamics (5) and to the following inequalities:

$$\begin{aligned} y_{\min} \le y(k+i|k) \le y_{\max}, \ & i = 1, \ldots, P, \ k \ge 0 \\ u_{\min} \le u(k+i|k) \le u_{\max}, \ & i = 0, \ldots, M-1, \ k \ge 0 \\ | \Delta u(k+i|k) | \le \Delta u_{\max}, \ & i = 0, \ldots, M-1, \ k \ge 0 \\ | \Delta u(k+i|k) | = 0, \ & i = M, \ldots, P-1, \ k \ge 0 \end{aligned} \tag{7}$$

with $Q_i \in \mathrm{R}^{p \times p}$, $R_i \in \mathrm{R}^{m \times m}$, $S_i \in \mathrm{R}^{m \times m}$, $\Delta u \in \mathrm{R}^{m}$ is the control increment vector and $r \in \mathrm{R}^{p}$ is the reference signal.

Given the convexity of the optimisation problem above (quadratic objective function and linear constraints) any particular solution is a global optimum and, hence, the open-loop optimal control problem can be restated as a quadratic programming problem (8) and (9).

$$\textit{minimise} \qquad J(\Delta \tilde{u}) = h^{T} \Delta \tilde{u} + \frac{1}{2} \Delta \tilde{u}^{T} H \Delta \tilde{u} \tag{8}$$

$$\textit{Subject to} \qquad A^{T} \Delta \tilde{u} \le b \tag{9}$$

where $A \in \mathrm{R}^{mM \times (4mM + 2pP)}$, $b \in \mathrm{R}^{(4mM + 2pP)}$ and $\Delta \tilde{u} \in \mathrm{R}^{mM}$ is denoted the extended control increment vector over the control horizon. The cost function gradient $h \in \mathrm{R}^{mM}$ and its Hessian $H \in \mathrm{R}^{mM \times mM}$ are given by:

$$\begin{aligned} h_l^T = 2 \Bigg\{ & x_o^T \left[\sum_{i=l-1}^{P-1} \left(\Xi \Phi^{i+1} \right)^T Q_{i+1} \sum_{q=0}^{i-l+1} \Xi \Phi^q \right] \Gamma \\ & + [\Gamma u_o + h]^T \sum_{i=l-1}^{P-1} \left[\sum_{q=0}^{i} \left(\Xi \Phi^q \right)^T Q_{i+1} \sum_{s=0}^{i-l+1} \Xi \Phi^s \right] \Gamma \\ & - \left[\sum_{i=l-1}^{P-1} \left(r(i+1) \right)^T Q_{i+1} \sum_{q=0}^{i-l+1} \Xi \Phi^q \right] \Gamma + u_o^T \sum_{i=l-1}^{P-1} R_i \Bigg\}, \\ & (l = 1, \ldots M) \end{aligned} \tag{10}$$

$$\begin{aligned} H_{ll} = 2 \Bigg\{ & \Gamma^T \sum_{i=0}^{P-l} \left[\sum_{q=0}^{i} \left(\Xi \Phi^q \right)^T Q_{i+1} \sum_{q=0}^{i} \Xi \Phi^q \right] \Gamma \\ & + \sum_{i=l-1}^{P-1} R_i + S_{l-1} \Bigg\}, \quad (l = 1, \ldots, M) \end{aligned} \tag{11}$$

$$H_{lp} = 2\left\{ \Gamma^T \sum_{i=l-1}^{P-1} \left[\sum_{q=0}^{i-l+1} \left(\Xi \Phi^q \right)^T Q_{i+1} \sum_{q=0}^{i-p+1} \Xi \Phi^q \right] \Gamma + \sum_{i=l-1}^{P-1} R_i \right\}, \quad \left(p = 1, \ldots, M;\ p \neq l \right) \tag{12}$$

In order to prevent from the effect of model/plant mismatches, which are responsible for persistent offsets, we propose in the present paper the incorporation of a pre-filter (compensator) within the control loop according to Figure 3.

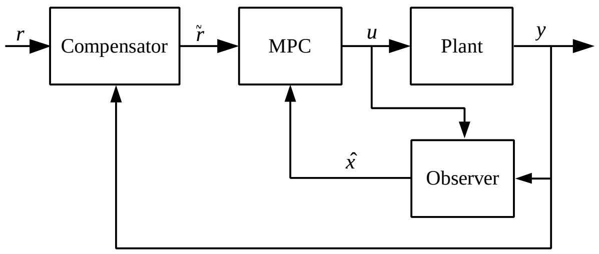


Figure 3: Extended MPC structure.

This element described altogether by (13) works somehow as an integrator where the control error is previously weighted by a decreasing exponential, as a function of the current error, before being added up to the cumulative error. By affecting the control error with a factor like this, it enables that only in a small neighbourhood of the system's current set-point the contribution to the cumulative error is significant. Next, this summation is added up to the true reference signal before being supplied to the MPC structure.

Additionally, the compensator should be able to reset the cumulative error whenever a changing in the true reference takes place, to prevent the MPC structure from receiving an unintended compensated reference, which could ultimately be responsible for windup outcomes. This task is here accomplished by affecting the previous cumulative error $J(k-1)$ with a Dirac delta function, d, having as argument the difference between the current and next set-points.

$$\begin{aligned} &J(k) = J(k-1)\, d\left(r(k+1) - r(k)\right) + e(k) \exp\left(-a\, \left|e(k)\right|\right) \\ &\tilde{r}(k+i) = r(k+i) + J(k), \quad i = 1, \ldots, P \end{aligned} \tag{13}$$

where $e(k)$ is the current tracking error and $a \in \mathrm{R}^+ \setminus 0$ is the coefficient of the exponential function. The tuning of this parameter is performed so that only very small absolute errors have significant contribution on the cumulative error.

4 EXPERIMENTS

To assess the feasibility of the proposed extended model predictive control strategy we have carried out some experiments on a bench three-tanks system. The set of experiments included a modelling stage by means of state-space neural network identification procedure and some control studies devoted to tracking and rejection to non-stochastic and unknown output disturbances.

4.1 Process Description

The bench three-tanks system (Figure 4) consists of three plexiglas cylindrical tanks with identical cross-section supplied with distilled water, in the case, whose liquid levels, h_1, h_2 and h_3, are measured via piezoresistive transducers. The tank T_3 is connected to the other two tanks by means of circular cross section pipes provided with manually adjustable ball valves. In tank T_2 is located the main outlet of the system, which is connected to the collecting reservoir through a circular cross-section pipe provided with an outflow ball valve, as well. Additionally, at each tank lies another connection to the reservoir, enabling the injection of particular disturbances under the form of leaks. Two diaphragm pumps are available for pumping distilled water from the reservoir to T_1 and T_2 tanks.

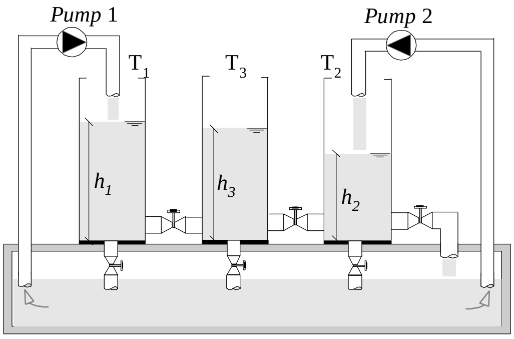
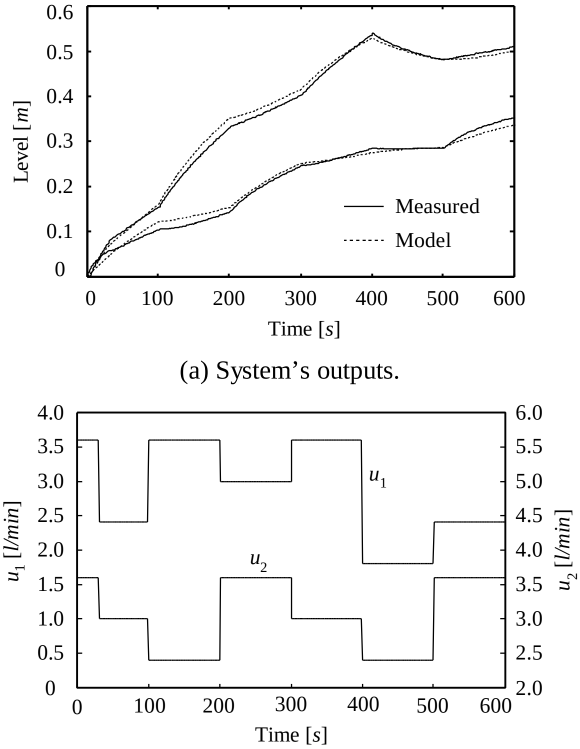


Figure 4: The three-tanks system schematics.

4.2 Results

For identification purposes, open-loop experiments have been carried out on the bench process in order to collect data to be used in the parameter estimation stage. In these experiments, step and pseudo random binary signals were applied to the system, with a sampling interval chosen as 1 second. Two of the collected records were picked for training the neural network and for subsequent cross-validation.

The structure of the neural network used for modelling purposes (Figure 1) is characterised by two neurons both in the input and output layers and three neurons in the hidden layer. After being trained, this neural model was able to predict fairly well the behaviour of the three-tanks system, as can be inferred from figures below.



(a) System’s outputs.

(b) Input signals.

Figure 5: Validation data set.

This non-linear neural predictor, forms thus a seed for all local instantaneous linear discrete-time models used within the model-based predictive control framework, which are extracted at each sampling time by means of Taylor series expansion.

For control purposes of the three-tanks system, the on-line open-loop constrained optimal control problem described by (8) and (9) is solved with a sampling of 1 second and choosing the prediction horizon as $P = 3$ time steps and the control horizon as $M = 1$ time steps. The weighting matrices in the objective functional (6) were chosen as:

$$Q_i = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \quad i < P \ ; \quad Q_P = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix} \tag{14}$$

$$R_i = \begin{bmatrix} 10^{-4} & 0 \\ 0 & 10^{-4} \end{bmatrix}, \quad S_i = \begin{bmatrix} 10^{-3} & 0 \\ 0 & 10^{-3} \end{bmatrix}, i \le P$$

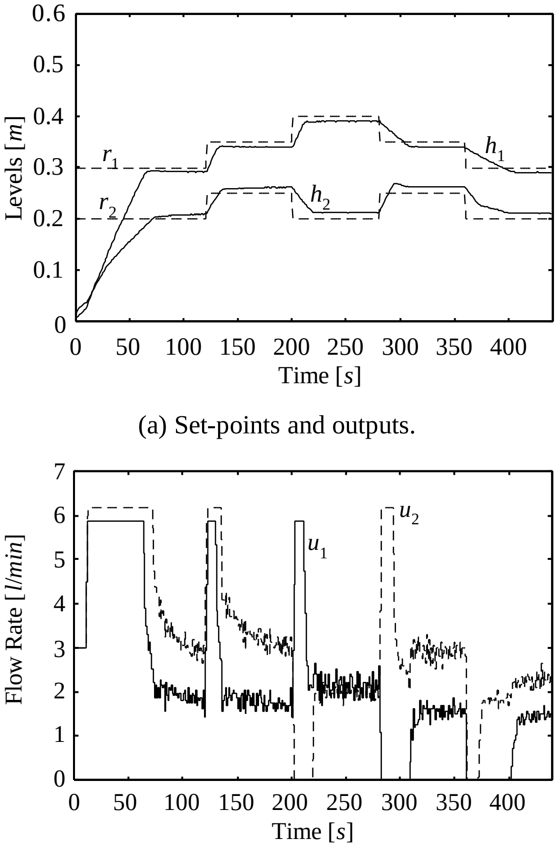
Given the existence of physical bounds on the flow rates supplied by the existing pumps and a finite maximum level for each of the tanks and, additionally, assuming upper and lower bounds in the control actions increments we impose the following constraints in solving the open-loop optimisation problem:

$$[0 \quad 0 \quad 0]^T \le x \le [0.6 \quad 0.6 \quad 0.6]^T \ m$$

$$[0 \quad 0]^T \le u \le [5.85 \quad 6.16]^T \ l/min \tag{15}$$

$$|\Delta u| \le [0.96 \quad 0.96]^T \ l/min$$

For assessing the true capabilities of the standard LIMPC without the incorporation of any offset compensation, a free-disturbance tracking experiment was conducted on the bench three-tanks system being the results plotted in Figure 6. As can be observed from this figure, the standard LIMPC scheme though guaranteeing a stable response with no constraints violation is not very impressive, at all, in terms of static offsets. These remnant small magnitude deviations are attributed mainly to modelling errors and also, most likely, with a comparative lesser contribution, to model degradation stemming from the first order Taylor expansion.

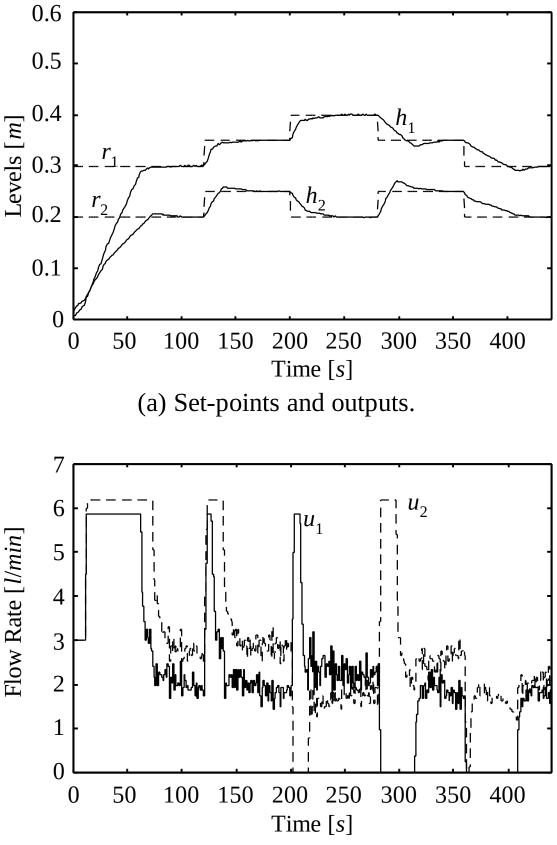


(a) Set-points and outputs.

(b) Control actions.

Figure 6: Standard model-based predictive control.

In order to prevent from static offsets or, at least, mitigate its magnitude, a pre-filter was incorporated into the standard LIMPC structure, according to Figure 3. Experimental results choosing the compensator's exponential coefficient as $a = 150$ and considering the same controller's parameters as those used in the standard LIMPC scheme are displayed in Figure 7. As it is clearly observed, the incorporation of an offset compensator into the control loop, by changing appropriately the reference signals supplied to the standard LIMPC structure contributes decisively to remove static offsets from the control system in a smooth way and in a finite time, as soon as the plant is brought to the set-point's neighbourhood. Furthermore, since control actions are carried out entirely by the standard model predictive controller violations of constraints are not allowed, even if the manipulated reference were laying outside the admissible region, which might occur for reference signals close to the corresponding bounds. On the other hand, windup behaviours in the manipulated references are not expected because changes in the reference signals are not very significant when the system's outputs are far from its set-points, due to the decreasing exponential factor contribution.

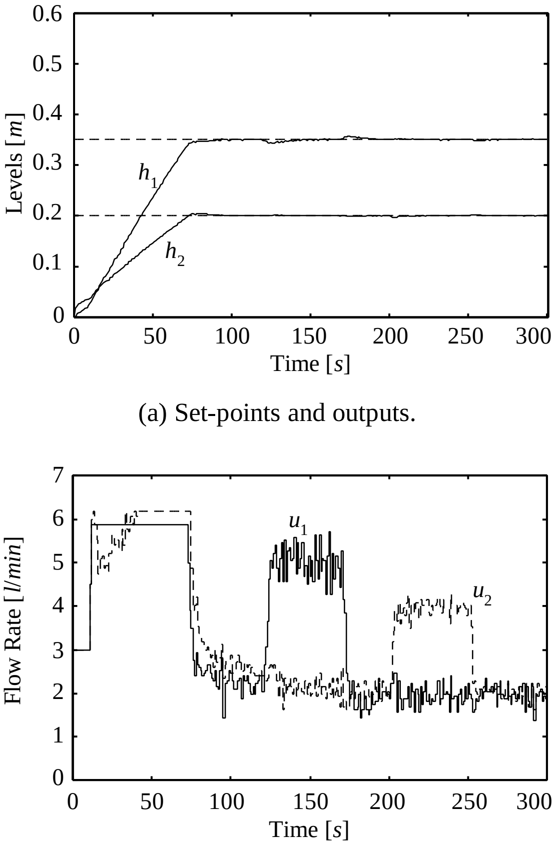


(a) Set-points and outputs.

(b) Control actions.

Figure 7: Extended model predictive control– tracking.

Next, to verify the control system's robustness to non-stochastic output disturbances we have carried out an experiment where two leaks were injected, not simultaneously, on tanks T_1 and T_2. Thus, at time 120 seconds, the ball valve in the circuit that connects tank T_1 to the bottom reservoir was partially open (30°) up to time 170 seconds, when it was closed totally and, by this way ceasing the disturbance. Next, from instant 200 seconds up to 250 seconds a similar procedure was carried out on tank T_2 and so an output disturbance was injected in this particular tank. In both cases, the set-points were kept unchanged over the entire experiment. As can be observed in Figure 8, the proposed control scheme is quite good in rejecting non-stochastic outputs disturbances, regardless the output being disturbed, bringing the system back to the set-point without any oscillations and, in addition, with no constraints violations either in the liquid levels or in the control actions, as would be expected since the main control structure is implemented by the LIMPC. However, for larger magnitudes of the injected disturbances it is expected a steady-state offset imposed by the existing bounds on the flow rates pumped by both pumps.



(a) Set-points and outputs.

(b) Control actions.

Figure 8: Extended model predictive control– disturbance rejection.

5 CONCLUSIONS

In this paper we have presented a recurrent neural model-based predictive control scheme assuring free or, at least, negligible steady-state offsets, in case of noisy systems, In order to mitigate steady-state offsets arising in standard MPC schemes due to model/plant mismatches and to model degradation in the linearisation step, a pre-filter with an integral behaviour is incorporated into the control loop. As control errors are weighted by a decreasing exponential, as a function of the errors themselves before entering the integrator section, only small errors play a significant role in changing appropriately the reference signal provided to the MPC unit.

Results from experiments have shown that the incorporation of the proposed pre-filter into the control loop together with the MPC scheme ensures indeed good tracking performances, despite modelling errors and, in addition, is also able to reject non-stochastic output disturbances in a very satisfactory way. This provides some insights on the proposed control system's robustness.

Acknowledgements

This work was partially supported by the ALCINE and the PRAXIS/P/EEI/14155/1998 programs.

References

[1] S. Quin and T. Badgwell, An overview of nonlinear model predictive control applications, *Nonlinear MPC Workshop.* Ascona, Switzerland, June 2-6, 1998.

[2] A. Propoi, Use of LP Methods for synthesizing sample data automatic systems, *Automation and Remote Control*, **24**, 1963, 837.

[3] J. Richalet, A. Rault, J. Testud and J. Papon, Model predictive heuristic control: Applictions to industrial processes, *Automatica*, **14**, 1978,413-428.

[4] C. Cutler and B. Ramaker, Dynamic matrix control- A computer control algorithm, *In Proc. AIChE 86th National Meeting*, 1979, Houston, Texas.

[5] A. Alamir and G. Bornard, Stability of a truncated infinite constrained receding horizon scheme: the general discrete nonlinear case, *Automatica*, **31**, 1995, 1353-1356.

[6] M. Soroush and C. Kravaris, MPC formulation of GLC, *AIChE Journal*, 42, 1996, 2377-2381.

[7] H. Chen and F. Allgöwer, A quasi-infinte horizon nonlinear model predictive control scheme with guaranteed stability, *European Control Conference, ECC*'97, 1997.

[8] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P.-Y. Glorennec, H. Hjalmarsson and A. Juditsky, Nonlinear black-box modelling in system identification: a unified overview, *Automatica*, **31**, 1995, 1691-1724.

[9] K. Narendra, Neural networks for control: Theory and practice, *Proc. IEEE*, **84**, 1996, 1385-1406.

[10] K. Funahashi, On the approximate realization of continuous mappings by neural networks, *Neural Networks*, **2**, 1989, 183-192.

[11] M. Garzon and F. Botelho, Dynamical approximation by recurrent neural networks, *Neurocomputing*, **29**, 1999, 25-46.

[12] K. Narendra and K. Parthasarathy, Identification and control of dynamical systems using neural networks, *IEEE Trans. On Neural Networks*, **1**, 1990, 4-27.

[13] E. Sontag, Neural networks for control, *In Essays on Control: Perpectives in the theory and its applications* (H. Trentelman and J. Willens, eds.), Birkhauser, Boston, 1993, 339-380.

[14] K. Hunt, D. Sbarbaro, R. Zbirowski and P. Gawthrop, Neural networks for control system – A survey, *Automatica*, 28, 1992, 1083-1112.

[15] I. Rivals and L. Personnaz, Black-box modelling with state-space networks, *Neural Adaptive Control Technology, R. Zbikowski and J. Hunt eds., World Scientific*, 1996, 237-264.

[16] M. Hagan, H. Demuth and M. Beale, *Neural Network Design* (PWS Publishing Company, Boston, 1996).

[17] M. Hagan and M. Menhaj, Training feedforward networks with the Marquardt algorithm, *IEEE Transactions On Neural Networks*, **5**, 1994, 989-993.