Adaptive Neural Predictive Techniques for Nonlinear Control

Nicolae Constantin

Bucharest Polytechnic University

Department of Automatic Control and Systems Engineering

313 Spl. Independentei Blvd., 77206, Bucharest

ROMANIA

E-mail : nicu@lnx.cib.pub.ro

Abstract: Predictive techniques based on neural networks are investigated in an adaptive structure for on-line control of a process exhibiting nonlinearities and typical disturbances. The method proposed consists of a novel identification technique based on additional memory adaptation and an efficient implementation of the predictive control, based on a nonlinear programming method. A forced circulation evaporator was chosen as a realistic nonlinear case study for the techniques discussed in the paper.

Keywords: predictive control, nonlinear models, neural networks, on-line control.

1. Introduction

One of today's trends in modern control is finding faster, a more reliable solution for highly nonlinear control problems in constrained environments. Many industrial processes exhibit a highly nonlinear behaviour across the operating range, they are often subjected to physical constraints in their process operation and they often have to follow safety limitations and strict environmental regulations.

The design of a model-based predictive control [2, 12] system relies heavily upon an explicit mathematical model of the system under control. In many cases such a model is very difficult to find due to, for instance, lack of knowledge about the system or the presence of strong nonlinear dynamics in the behaviour of the system. For this reason conventional control systems are usually based on a linearized and highly simplified mathematical model of the system under control. The use of simplified models in model based control usually leads to a decrease of performance of the overall system.

In [10] it was proposed the enhancement of model-based control strategies, such as indirect adaptive control, by replacing the linear models with nonlinear representations, based on neural networks (NN). It has been shown that predictive control techniques could be improved by including a nonlinear model of the plant. In particular, the dynamic matrix control (DMC) was extended by using a nonlinear predictive model of the process, based on feedforward neural networks.

Neural networks are particularly appropriate for multivariable applications, where they can readily characterise the interactions between different inputs and outputs [8].

The goal of this paper is to show that, provided with appropriate adaptation techniques and control structures, neural networks can be used to adaptively control a wide range of nonlinear processes at a useful level of performance.

The organisation of the paper is as follows. In Section 2 some aspects of the predictive control strategy are presented. Identification structures with NN and the novel adaptation technique AMA is proposed and analysed in section 3. Section 4 shows some of the simulation results of adaptive control for a multivariable nonlinear process. Conclusions are presented in Section 5.

2. Nonlinear Predictive Control

The complexity of developing a phenomenological model for the process and the nonlinearities of its dynamics, make very attractive the use of an artificial neural network for the identification and predictions of the plant outputs. Fixed NN models, identified off-line have been used with various model-based control structures [13].

Prediction using neural networks

The prediction algorithm uses the output of the plant's model to predict the plant's dynamics to an arbitrary input from the current time k to some future time k+n. This is accomplished by time shifting equations for $y_n(k)$ and $net_j(k)$ by n resulting in

$$y_n(k+n) = \sum_{j=1}^{nh} w_j f_j\big((net_j(k+n)\big) + b \quad (1)$$

and

$$net_j(k+n) = \sum_{i=0}^{nu} w_{j,i+1} \begin{cases} u(k+n-i), n-N_u < i \\ u(n+N_u), n-N_u \geq i \end{cases} + \sum_{i=1}^{\min\{n,ny\}} w_{j,nu+i+1} y_n(k+n-i) + \\ + \sum_{i=n+1}^{ny} w_{j,nu+i+1} y(k+n-i) + b_j \quad (2)$$

where $f_j(.)$ is the output function for the j^{th} node of the hidden layer, $net_j(n)$ is the activation level of the j^{th} node's output function, nh is the number of hidden nodes in the hidden layer, w_j is the weight connecting the j^{th} hidden node to the output node ,$w_{j,i}$ is the weight connecting the i^{th} input node to the j^{th} hidden node, b_j the bias on the j^{th} hidden node, b the bias on the output node. The first summation of (2) breaks the input into two parts represented by the conditional.

Using quadratic cost function and the predicted model it is possible to calculate the optimal control strategy for a nonlinear model predicted by using NN. The cost function in predictive control is chosen as:

$$J = \sum_{j=N_1}^{N_2} \left[r(k+j) - y_n(k+j) \right]^2 + \sum_{j=1}^{N_u} \lambda(j) \left[\Delta u(k+j) \right]^2 \quad (3)$$

where r is the required process output, y_n is the NN model output, u is the process input, N_1 and N_2 define the prediction horizon, N_u the control horizon and λ is a sequence of control weighting factors. A suitable choice for N_1 is to make it equal to the process delay between input and output. N_2 is then set to define the prediction horizon beyond this point and it represents the number of time-steps in the future for which the process response is recursively predicted as a result of the proposed control action sequence u(k), ..., u $(k+N_u-1)$ executed over the control horizon N_u . Control actions after the control horizon are held constant equal to the value $u(k+N_u-1)$. With Newton-Raphson (NR) method, J is minimized iteratively to determine the best

$$\underline{u}(n) = \left[u(k+1)\, u(k+2) \ldots u(k+N_u) \right]^T \quad (4)$$

The improved convergence rate of NR is computationally costly, but is justified by the high convergence rate. The NR update rule for $\underline{u}(n+1)$ is:

$$\underline{u}(n+1) = \underline{u}(n) - \left(\frac{\partial^2 J}{\partial \underline{u}^2}(n) \right)^{-1} \frac{\partial J}{\partial \underline{u}}(n) \quad (5)$$

In order to avoid the computation of the inverse of the Hessian matrix, equation (11) is rewritten in the form of a system of linear equations: $\frac{\partial^2 J}{\partial \underline{u}^2}(n)\big(\underline{u}(n+1) - \underline{u}(n)\big) = \frac{\partial J}{\partial \underline{u}}(n)$ which is solved for $x = \underline{u}(n+1) - \underline{u}(n)$. When solving for x, calculation of each element of the Jacobian and Hessian is needed for each NR iteration.

3. Adaptive Predictive Control

A major advantage of the predictive control strategy is that the process input is optimally computed to take into account of future predictions of the process response up to a defined horizon. This enables the scheme to anticipate future process outputs, which achieves smooth transient responses to set-point changes and the ability to take early corrective action against disturbances. The adaptive scheme (Fig. 1) is obtained by combining predictive control with the EM adaptation method. This structure has a number of attributes that make it an appealing strategy to adopt in comparison to other neural-network control schemes.

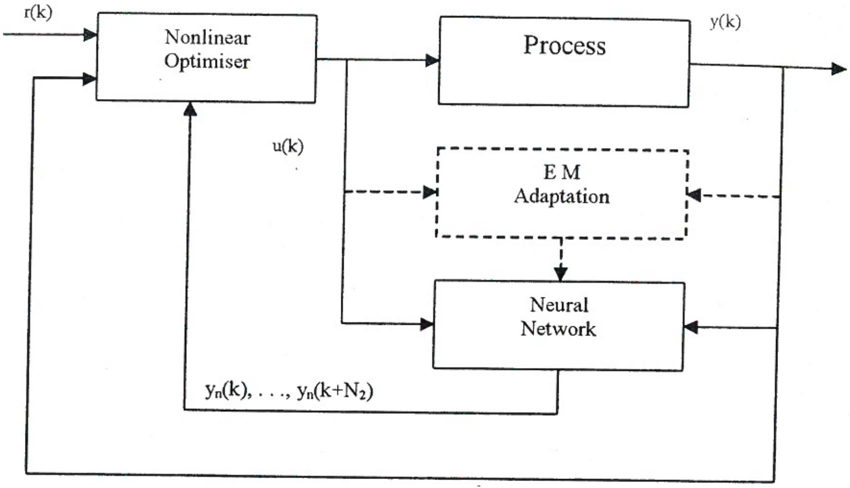

Figure 1.

3.1. Additional-Memory Adaptation Technique

A Multiple-Input Multiple-Output (MIMO) model suitable for identification can be expressed in terms of θ(k) , the set of unknown parameters, past values of the inputs and outputs, and modelling errors e(k) :

$$y(k) = f(\theta(k), y(k-1), \ldots, y(k-n_y), \quad u(k-1), \ldots, u(k-n_u)) + e(k) \tag{6}$$

At time step k each weight w_{ji} , connecting the input of node i on each layer, to the output of node j on the preceding layer, is updated using:

$$\begin{aligned} w_{ji}(0) &= \gamma_{ji} \\ w_{ji}(k) &= w_{ji}(k-1) + \alpha e_i(k) + \beta(w_{ji}(k-1) - w_{ji}(k-2)) \end{aligned} \tag{7}$$

where $\alpha \in R^+$ is the learning rate , $0 < \beta < 1$ is the momentum factor, $e_i(k)$ is the back propagated error value corresponding to the input to the sigmoid function calculated for node i during the presentation of the pattern θ(k) derived from process time k. The random variable γ_{ji} is independently selected for each weight from a distribution with zero mean and small variance.

A distinction should be made between iteration q and time step k. The updated weight w_{ji} (q,k) is calculated using the extended recursive equation set below:

$$\begin{aligned} w_{ji}(0,0) &= \gamma_{ji} \\ w_{ji}(0,0) &= w_{ji}(1, k-1) \\ w_{ji}(q,k) &= w_{ji}(q-1,k) + \alpha e_i(k) + \beta(w_{ji}(q-1,k) - w_{ji}(q-2,k)) \end{aligned} \tag{8}$$

The method proposed retains some process patterns that can represent an approximation to the nonlinear process dynamics. The additional memory (AM) accepts a new pattern θ(k) from the process at each time step k and discards the oldest pattern $\theta(k-n_p)$. The recursive equation set can be written as follows:

$$\begin{aligned} w_{ji}(0,0) &= \gamma_{ji} \\ w_{ji}(0,0) &= w_{ji}(1, k-1) \\ w_{ji}(q,k) &= w_{ji}(q-1,k) + \alpha e_i(p) + \beta(w_{ji}(q-1,k) - w_{ji}(q-2,k)) \end{aligned} \tag{9}$$

where: l is a cycle counter for n_c cycles of the pattern memory which are performed at time step k , p indicated the time step origin of the pattern selected from the AMA is chosen in random order from its index range $(k - n_p + 1 , k)$. At each selection of p and execution of the recursion, the iteration count q is incremented, up to $n_c n_p$.

3.2. AM parameters

The learning rule is repeated n_p x n_c times at each time-step. The parameter n_p determines the length of moving average effect on the patterns. It is set as a balance between speed of adaptation and accuracy of model. It is analogous to the forgetting factor commonly used in linear adaptive control theory [9]. Using various situation typical values of n_c and n_p for various nominal conditions and objectives have been used in simulations. In the case study simulations we demonstrate greatly enhanced identification performance in adaptive control structures and also the possibility of starting identification and control at the same time with no a priori knowledge of the plant.

4. Simulation Results

Neural control techniques have already been introduced in many industries, in particular to the chemical and biochemical processes [5, 9]. Here is considered a complex process simulation of a forced circulation evaporator system. The process diagram is presented in figure 2. The variables, their description, normal operation states with measurements units are described in table 1. The simulation equations are presented in [11]. The resulting process simulation is dynamically nonlinear, unstable, and multivariable with strong interactions. An input-output neural model with n_y=2 and n_u=2 was selected. The final network topology was selected to be 12-20-3 . The EM parameters were set to n_c=20 si n_p=50.

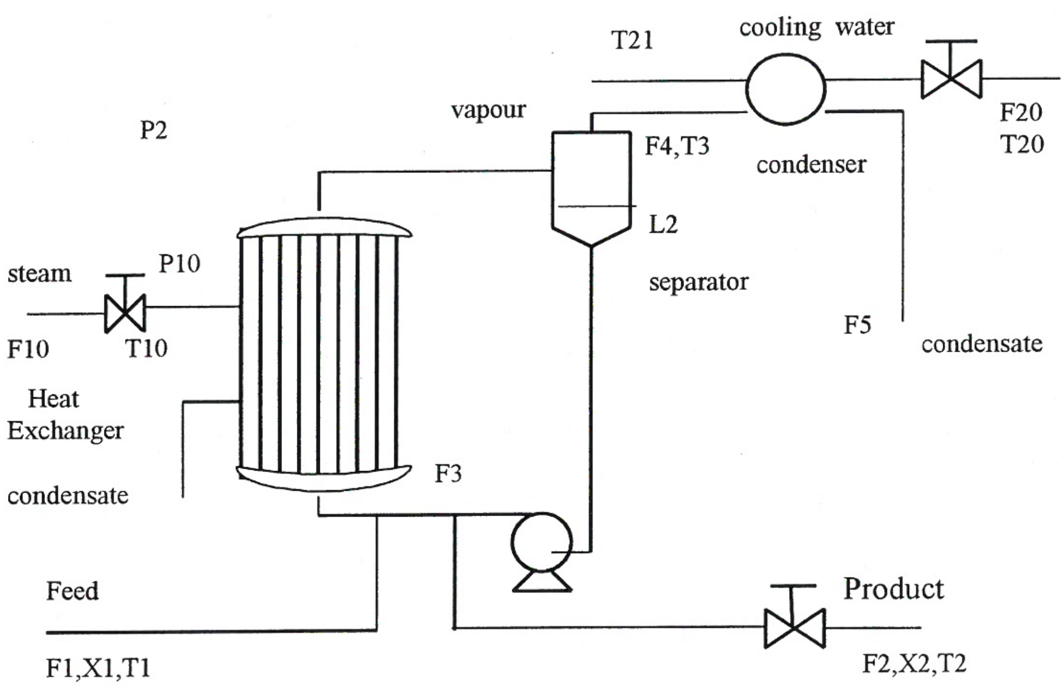

Figure 2. Forced Circulation Evaporator

Table 1. Evaporator process variable description

Variable	Description	Nominal Value
F1	feed flowrate	10.0 kg/min
F2	product frowrate	2.0 kg/min
F3	circulating flowrate	50.0 kg/min
F4	vapour frowrate	8.0 kg/min
F5	condensate flowrate	8.0 kg/min
X1	feed composition	5.0 %
X2	product composition	25.0 %
T1	feed temperature	40.0 °C
T2	product temperature	84.6 °C
T3	vapour temperature	80.6 °C
L2	separator level	1.0 meters
P2	operating pressure	50.5 kPa
F10	steam flowrate	9.3 kg/min
T10	steam temperature	119.9 °C
P10	steam pressure	194.7 kPa
Q10	heater duty	339.0 kW
F20	cooling water flowrate	208.0 kg/min
T20	cooling water outlet inlet temperature	25.0 °C
Q20	condeser duty	307.9 kW
T21	cooling water outlet temperature	46.1 °C

Test sequence

A sequence of test cases using the proposed method were conducted to investigate initial start-up adaptation and stabilization, setpoint following and disturbance rejection. The first stage of the test reflects the mode of start-up. After a period for adaptation and stabilisation of the process, a sequence of alternating setpoint step (for X2) set to 1 , 0.5, -0.5, -1, 1, -1 was introduced. Finally it was introduced a large (30%), unmeasured disturbance.

The performance of each of the cases and their individual components were compared using the Integrated Time-weighted Absolute Error (ITAE) measure. In this experiment the control rate-of-change weight was set to 0.3 ($\lambda = 0.3$). The plot of the three outputs is shown in Fig.3 for $N_2 = 5$ and $N_u = 3$, np = 40 and nc = 10.

The process was quickly stabilized and the measurements had varied at most by only 5, 6 and 20% respectively from their operating values. The 20 % variation of y_3 is associated with the separator level, the unstable state of the process. The closed-loop responses indicate very good control of all outputs.

The initial stabilization tends to have fewer, lower amplitude, slower cycles in comparison with the case when the rate-of-change weight was set to zero. Several others experiments were realised to investigate the effects of different values of the design variables within the control structure [3].

The results obtained were in accord with those expected from using a predictive control methodology. It is seen that with no weighting of the control input good set-point tracking is achieved; however, the control input exhibits large fluctuations and would cause undue actuator wear in practice. For larger values of λ , movement of the manipulated variable is dampened at the expense of poorer set-point tracking. It was observed that increasing the prediction horizon had a number of desirable effects. Good set-point tracking was maintained, whilst the control effort was considerably reduced.

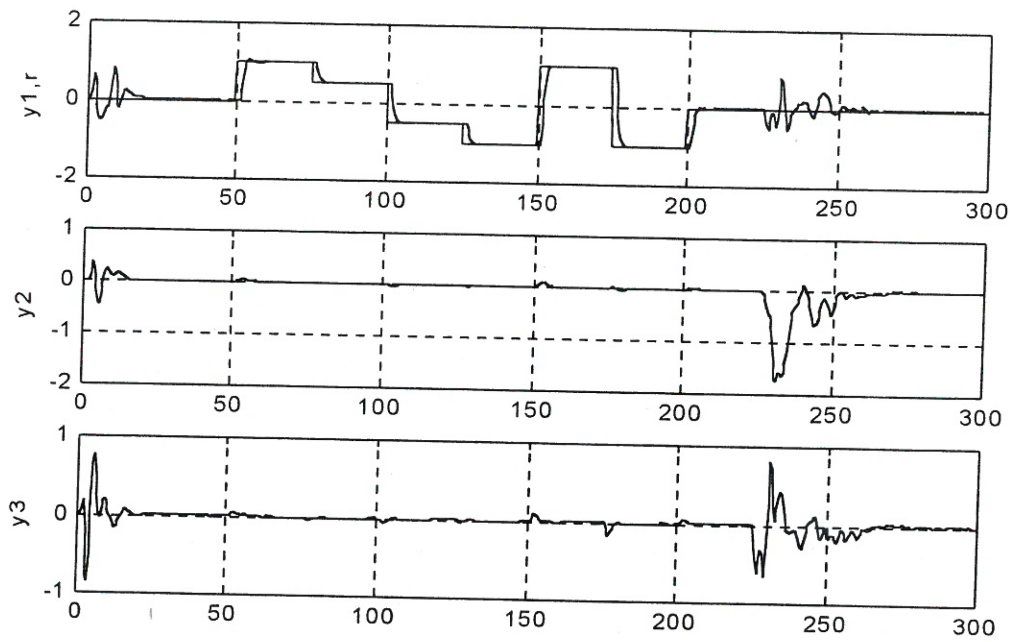

Figure 3. Process Response with $N_u = 3$, $N_2 = 5$, $n_p = 40$, $n_c = 10$.

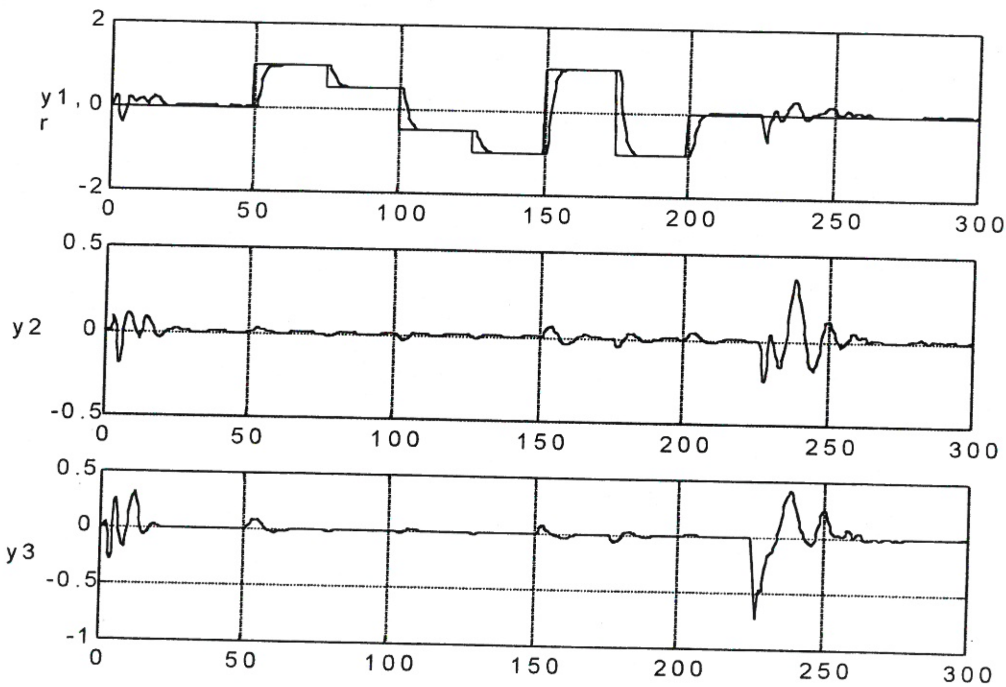

Figure 4. Process Response with $N_u = 2$, $N_2 = 3$, $n_p = 40$, $n_c = 10$.

These results demonstrate that improvements can be achieved with neural network in practice. The presented method is better than the other approaches due to fast adaptation and ability to address process nonlinearity. For time-varying processes, adaptation is the only means by which the performance of the controller can be maintained.

The values of the parameters were decreased in order to reduce the computation effort. The control horizon and stack size had the main contribution on this reduction but also the performance suffers degradation. A trade-off between the computational effort and performances was obtained. The number of cycles for each step of EM algorithm could be reduced but the obtained performances were not improved significantly. The goal was to maintain a suitable level of performances in parallel with a reduction of computational effort. Based on this approach it could be obtained good results with reasonable computational effort.

5. Conclusions

This paper has discussed issues concerning the application of predictive techniques based on neural networks to the adaptive control of nonlinear processes. A novel identification method is proposed for the learning of nonlinear input-output mappings. This identification method is combined with a multi-step predictive control to provide an adaptive neural predictive control strategy. The proposed method is superior due to fast adaptation and ability to address process nonlinearities. It is a viable adaptive control approach which can achieve high performance when applied to nonlinear complex processes.

REFERENCES

1. CHEN, S. S.A. BILLINGS (1989). **Representation of nonlinear systems: The NARMAX model**. Int.J.Contr., 49(3), pp. 1013-1032.
2. CLARKE, D.W., C.MOHTADI, AND P.S.TUFFS (1987). **Generalized predictive control-Part1: The basic algorithm**. Automatica, 23, pp. 137-148.
3. CONSTANTIN, N. AND I. DUMITRACHE (1999). **Advanced Adaptive Techniques for Multivariable Nonlinear Processes**, Proceedings of the 14th IFAC World Congress, vol. I, pp. 174-180.
4. CONSTANTIN, N. AND I. DUMITRACHE (1997). **Multimodel Adaptive Control**. Proc. Inter. Conf. CSCS11, 1, pp. 259-264, Bucharest.
5. GOMM, J.B., J.T. EVANS AND D. WILLIAMS (1997). **Development and performance of a neural-network predictive controller**. Control Eng. Practice, vol. 5, No.1, pp. 49-59.
6. HORNIK, K., M. STINCHOMBE AND H. WHITE (1989). **Multilayer feedforward networks are universal approximators.** *Neural Networks*, pp. 2, 359-366.
7. HUNT, K.J., G.W. IRWIN AND K. WARWICK (1995). **Neural Network Engineering in Dynamic Control Systems**. London: Springer-Verlag.
8. HUNT, K.J., D. SBARBARO, R. ZBIKOWSKI AND P.J. GAWTHROP (1992). **Neural network for control systems**. Automatica, 28(6), pp. 1083-1112.
9. MILLS, P.M., A.Y. ZOMAYA AND M.O. TADE (1996). **Neuro-Adaptive Process Control - A Practical Approach**. John Wiley & Sons.
10. NARENDRA, K.S. AND K. PARTHASARATHY (1990). **Identification and control of dynamical systems using neural networks**. IEEE Trans. on Neural Networks, 1(1), pp. 4-27.
11. NEWELL, R. AND P. Lee (1989). **Applied Process Control : A Case Study**. Englewood Cliffs, New Jersey: Prentice Hall.
12. RICHALET, J. (1992). **Observations on model-based predictive control.** Control Eng. Practice, Vol 8,38-41.
13. SAINT DONAT, J., N. BHAT AND T.J. MCAVOY (1991). **Neural net based model predictive control**. Int. J. Control, 54(6), 1453-1468.
14. WELLSTEAD, P.E. AND M.B. ZARROP (1991). **Self-tuning Systems.** New York:Willey.