ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 391-396

# Neuro-predictive control based self-tuning of PID controllers

Corneliu Lazar, Sorin Carari, Draguna Vrabie, Marius Kloetzer

"Gh. Asachi" Technical University of Iasi, Department of Automatic Control
Blvd. D. Mangeron 53A, 700050 Iasi, Romania
{clazar,dvrabie,scarari,kmarius}@ac.tuiasi.ro

Abstract: In this paper we present a new self-tuning procedure for PID controllers based on neuro-predictive control. A finite horizon optimal control problem is solved on-line, permitting to calculate the tuning parameters of the PID controller. The proposed method is implemented on a level-flow pilot plant and a comparison with conventional auto-tuning methods is also given.

## 1. Introduction

Although most of the industrial processes are complex nonlinear systems, they are still controlled with classical PID control structures, which are tuned to give good results only around a fixed operating point. Under these circumstances, in order to obtain the optimal response over the entire operating range, on-line adaptation or self-tuning of the controller is required, and several methods have been proposed in the last decade, e.g. [1], [2], [3].

In [1], the existent types of adaptive techniques are classified based on the fact that if the process dynamics are varying, then the controller should compensate these variations by adapting its parameters. There are two types of process dynamics variations: predictable and unpredictable. The predictable ones are typically caused by nonlinearities and can be handled using a gain schedule, which means that the controller parameters are found for different operating conditions with an auto-tuning procedure that is employed thereafter to build a schedule.

In this paper, a new self-tuning method for PID controllers designed to control processes with predictable dynamics variations is presented. The gain scheduling principle is replaced by using a neural network based model that is capable to capture the predictable dynamics variations of the process. The neural network model is also used to develop a neural structure that predicts the future control error caused by process dynamics variations. The controller tuning parameters are calculated solving a finite horizon optimal control problem that minimizes the predicted control error. Real-life experimental results are given for a level-flow pilot plant, which demonstrate the practical benefits of this self-tuning method.

## 2. Description of the self-tuning procedure

The proposed self-tuning approach is based on two parallel control structures (see Fig. 1) that are synchronized with the reference clock of the predictable dynamics

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 391-396

process in closed-loop with a PID controller.

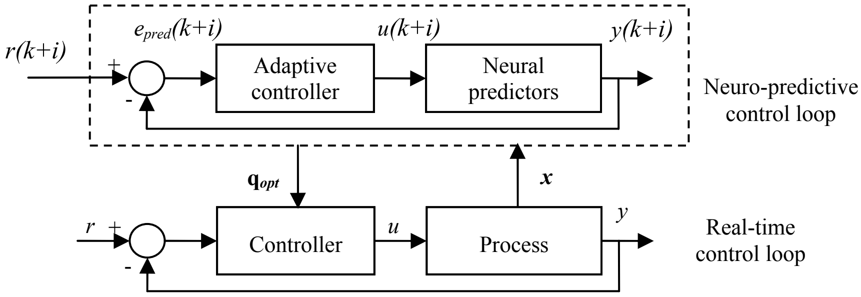


Figure 1: Neuro-predictive structure

The upper structure uses a predictive control loop consisting in a neural predictor and a PID controller with adaptive tuning parameters. The predictive structure, with the sampling rate $T_p$, works faster than the real-time control loop supplying the predicted control error over a finite future time horizon. The tuning parameters are calculated at each sample time instant through the minimization of the predicted control error and the obtained values are used to update the tuning parameters of the real-time control loop. Thus, the controller parameters are adapted based on the predictive optimization of the control system behavior and the desired performances can be achieved over the entire operating range.

### 2.1 Process neural model

Neuro-predictive control loop contains a neural network model, which models the real process with predictable dynamic variations. The use of neural networks for nonlinear process modeling and identification is justified by their capability to approximate unknown non-linear systems. A nonlinear model that includes a large class of non-linear processes is the following NARMAX model:

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

where *f(.)* is some nonlinear function, *d* is the dead time , *n* and *m* are the orders of the nonlinear system model, *u* and *y* being the input and the output of the process. A neural network based model, NNARMAX, corresponding to the NARMAX model, may be obtained by adjusting the weights of a multi-layer perceptron architecture with adequately delayed inputs [4]. The neural network output will be given by:

$$y(k) = f^N[\mathbf{u}(k-d-1), \mathbf{y}(k-1)], \tag{2}$$

where $f^N$ denotes the input-output transfer function of the neural network, which replaces the non-linear function *f* in (1) and, $\mathbf{u}(k\text{-}d\text{-}1)$ and $\mathbf{y}(k\text{-}1)$ have the following structure:

$$\begin{aligned} \mathbf{u}(k-d-1) &= [u(k-d-1)u(k-d-2)...u(k-d-m)]^T \\ \mathbf{y}(k-1) &= [y(k-1)y(k-2)...y(k-n)]^T \end{aligned} . \tag{3}$$

For a two layer network, the following expression is obtained from equation (2):

$$y(k)=\sum_{j=1}^{n} w_j\sigma_j(\mathbf{w}_j^u\mathbf{u}(k-d-1)+\mathbf{w}_j^y\mathbf{y}(k-1)+b_j)+b\ , \tag{4}$$

where $n$ is the number of neurons in the hidden layer, $\sigma_j$ is the activation function for the $j$-th neuron from the hidden layer, $\mathbf{w}_j^u$ the weight vector for the $j$-th neuron with respect to the inputs stored in $\mathbf{u}(k\text{-}d\text{-}1)$, $\mathbf{w}_j^y$ the weight vector for the $j$-th neuron with respect to the inputs stored in $\mathbf{y}(k\text{-}1)$, $b_j$ the bias for the $j$-th neuron from the hidden layer, $w_j$ the weight for the output layer corresponding to the $j$-th neuron from the hidden layer and $b$ the bias for the output layer. Such structures with a single hidden layer are considered satisfactory for most of the cases.

Since all the industrial processes are working in closed-loop, a closed-loop identification method has been used to obtain the neural model of the process. In order to capture all the nonlinear dynamics of the process, the training data had to be attained around several different operating points such that the entire variation range of the process output to be covered. For this reason, a stepwise reference was chosen and then summed with a pseudo random binary signal generated with a shifting register [5].

### 2.2 Neuro-predictive control loop

In order to obtain the predictable dynamics variations at the time instants $k$, a neural predictor based on the neural-based model of the process was used. A sequential algorithm based on the knowledge of current values of $u$ and $y$ together with the neural network system model gives the $i$-step ahead neural predictor:

$$y(k+i)=\sum_{j=1}^{n} w_j\sigma_j(\mathbf{w}_j^u\mathbf{u}(k-d+i-1)+\mathbf{w}_j^y\mathbf{y}(k+i-1)+b_j)+b\ . \tag{5}$$

The future control $\mathbf{u}(k\text{-}d\text{+}i\text{-}1)$ from (5) is obtained running the neuro-predictive control loop. Thus, at time instant $k$, the predicted output $y(k+i)$ is determined, for $i=\overline{N_1,N_2}$ where $N_1$ and $N_2$ are the prediction horizons. If $T_p$ is the sampling time with which the predictive control loop operates, this must satisfy: $(N_2-N_1)T_p<<T$ .

Placing the neural model of the process to operate in the neuro-predictive control loop allows for transferring the current state $\boldsymbol{x}$ of the process to the neural predictor (Fig. 1) at each time instant $k$. Thus, at each time instant $k$, the predicted behavior of the process is obtained in the vector form:

$$\mathbf{y}_{pred}=[y(k+N_1)y(k+N_1+1)...y(k+N_2)]^T\ . \tag{6}$$

The process output $\mathbf{y}_{pred}$, predicted by the neural predictor, is used to calculate the predicted control error based on the controller set-point.

Considering the discrete form of a PID controller,

$$u(k)=u(k-1)+q_0e(k)+q_1e(k-1)+q_2e(k-2)\ , \tag{7}$$

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 391-396

and the model (5), yields the following equation for the predicted control error:

$$e_{pred}(k+i) = \left( \sum_{j=1}^{n} w_j \sigma_j (\mathbf{w}_j^u \mathbf{u}(k-d+i-1) + \mathbf{w}_j^y \mathbf{y}(k+i-1) + b_j) + b \right) - r(k+i), \quad (8)$$

where the vector $\mathbf{u}(k\text{-}d\text{+}i\text{-}1)$ is a function of the tuning parameters vector $\mathbf{q}=[q_0\ q_1\ q_2]$.

Minimizing the cost function

$$J = \frac{1}{2} \sum_{i=N_1}^{N_2} e_{pred}^2(k+i) \quad (9)$$

with respect to the variable $\mathbf{q}$, yields the optimal tuning parameters $\mathbf{q}_{opt}$, which compensate the process predictable dynamics variations. At the time instant $k$+1, the tuning parameters $\mathbf{q}_{opt}$ are transferred to the real-time control loop.

The self-tuning method has the advantage that it does not imply a pre-tune phase because the vector $\mathbf{q}_{opt}$ is available after the first sampling period.

## 3. Pilot plant studies

The neuro-predictive self-tuning approach for PID controllers was tested on a level-flow pilot plant. The schematic diagram of the pilot plant is presented in Fig. 2.

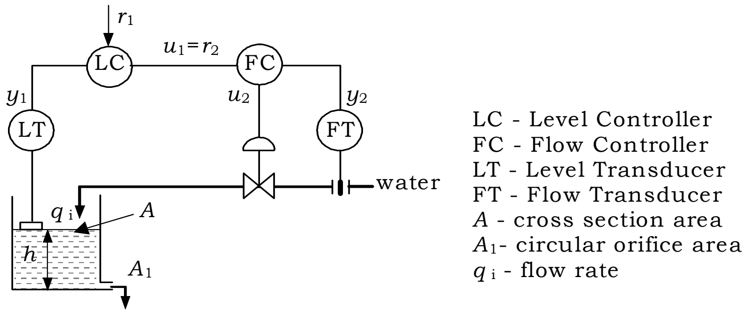


Figure 2: Schematic diagram of the level-flow pilot plant

The level is controlled using a cascade control structure that has as internal variable the feed water flow. The inner loop controls the feed water flow and rejects the disturbances caused by the pressure variations in the water pipe. The outer loop controller, a PI controller, yields the feed water flow reference signal $r_2$ for the inner loop based on the received water level signal. The tank and the inner loop represent the plant for the outer controller, which was modelled with a neural network in order to implement the developed self-tuning method for the level controller. The following non-linear model is available for the tank:

$$A\frac{dh}{dt} = q_i - C_d A_1 \sqrt{2gh} \ . \quad (10)$$

The model parameters are: $A$ = 203.4 cm$^2$, $A_1$ = 2.26 cm$^2$, $h_{max}$ = 13.5 cm, $C_d$ = 0.6.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 391-396

### 3.1 Neural model of the plant

In order to estimate the parameters of the neural model, a training sequence was built such that the process output explores its whole operating range. Thus, a stepwise reference summed with a pseudo random binary signal was applied to the real time control loop and, by monitoring the control signal $u_1$ and the process output $y_1$, a training sequence was obtained. Using the training sequence, a two layer neural network was trained off-line.

Model parameters $m$, $n$ and $d$ were estimated based on the collected input-output data and on the physical structure of the process (the inner flow loop and the tank). With a sampling rate of 2sec, it was found that the process has a delay $d$=2 and $m$=2, $n$=2. For the training and the validation of the neural network that models the process, the software instruments presented in [6] were used.

In Figure 3, the results of a closed-loop experimental validation of the neural model are plotted.

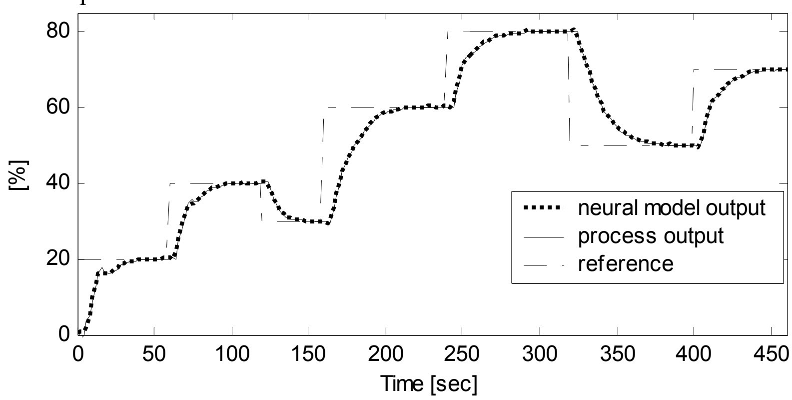


Figure 3: Process neural model validation

### 3.2 Adaptive procedure results

The parameters $q_0$ and $q_1$ of the PI controller were obtained by minimizing an objective function using an optimization approach. The cost function was taken as the mean squared prediction error over a finite prediction horizon, as in equation (9). The neural network model identified in the previous section was used to predict the future outputs. In order to avoid the saturation of the actuator, the minimization took into account the constraints imposed on the control input and controller parameters.

In order to do a fair comparison with the existing auto-tuning methods, the PI level controller was first tuned using the relay method of Astrom and Hagglund [3]. Figure 4 shows the resulting real-time closed-loop stepwise responses obtained with fixed tuning parameters of the PI controller and with continuous adaptation of tuning parameters based on the self-tuning method presented in this paper. In the same figure, the control inputs are depicted together with the tuning parameters of the PI controller.

As seen in the figure, the fixed parameters controller gave a sluggish control response. With the neuro-predictive tuning approach, the controller had continuous

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 391-396

adaptation of the tuning parameters resulting in a faster control.

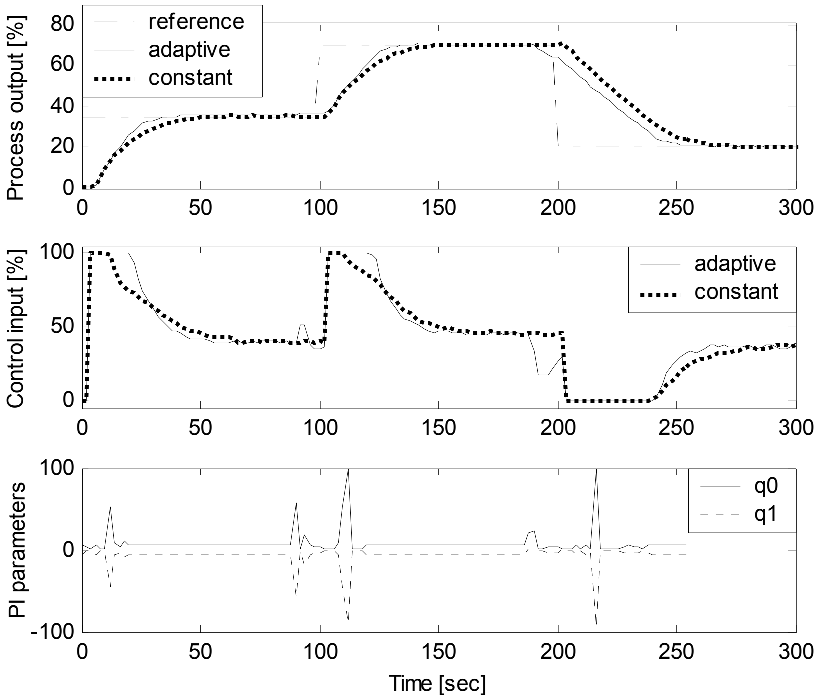


Figure 4: Comparative results and parameters for the adaptive discrete PI controller

## 4. Conclusions

A neuro-predictive control based self-tuning procedure for PID controllers has been developed. The main advantage of the method consists in the on-line adaptation of the controller parameters and in the possibility to track different process operating regimes. The proposed method has been implemented on a benchmark real-life system with good results and a comparison with a *classical* auto-tuning method for PID controllers has also been given.

## References

**1.** Hagglund T., Astrom K.J., Industrial Adaptive Controllers Based on Frequency Response Techniques, *Automatica*, vol.27, no.4, 599-609 (1991).
**2.** Cheng-Ching Yu, *Autotuning of PID Controllers*, Springer-Verlag, London (1999).
**3.** Astrom K.J. Hagglund T., *PID Controllers: Theory, Design and Tuning*, Instrument Society of America: Research Triangle Park, (1995).
**4.** Chen S, Billings S.A. and P.M. Grant, Nonlinear system identification using neural networks, *Int. J. Control*, vol 49, no.3 , 1013-1032 (1990).
**5.** Landau I.D., *System Identification and Control Design*, Prentice Hall, (1990).
**6.** Norgaard, M, *Neural Network Based System Identification Toolbox*, Technical Report 00-E-891, Department of Automation, Technical University of Denmark, (2000).