9th Symposium on Neural Network Applications in Electrical Engineering, NEUREL-2008
Faculty of Electrical Engineering, University of Belgrade, Serbia, September 25-27, 2008

Neural network control approach for a two-tank system

Mladen Majstorović, Ivan Nikolić, Jelena Radović, Goran Kvaščev, *Member, IEEE*

***Abstract*—This paper describes two different approaches in a two tank-system control using neural networks - the NARMA-L2 Control and the Model Reference Control. Knowing that the process control has had the most satisfying results using a standard PID controller, this paper compares the two approaches' results one to another but also every one of them with the PID controller's results. The goal was to increase the system response speed without heavily increasing the two other relevant parameters – the overshoot and the steady state error. All of the experiments, measurements and simulations were conducted in Matlab/RT Simulink.**

***Index Terms*—NARMA-L2, neural networks, reference control, two-tank system.**

I. Introduction

The aim of this paper is to present different structures of neural networks controllers with which we have tried to achieve a performance enhancement in the two-tank system control regarding the system response speed. In that process, we have not considered reducing the necessary computational time needed to train some of the neural networks, but only enhancing the results. The mathematical model of the two-tank system will be presented first, followed by a brief description of the two neural network controllers mentioned above, the NARMA-L2 and the Model Reference Control.

II. The two-tank system process and model

The two-tank system in question is actually the Amira DTS200 three-tank system without the third tank. The reason why the third tank was not used is to simplify the problem. Knowing that, the schematic description of the process is shown in the following figure.

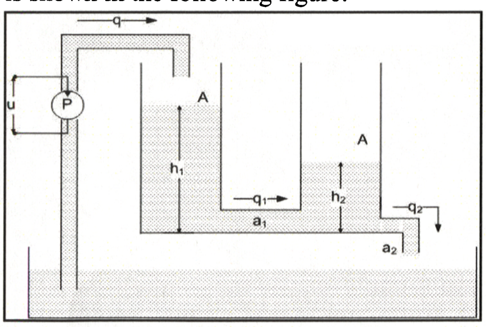

Fig. 1. The two-tank system

Authors are with the Faculty of Electrical Engineering, University of Belgrade, Serbia.

The symbols in the diagram are: P - the electrical pump, u – the pump's voltage, A – the tank's cross-section, q – the flow rate provided by the pump, q_2 – the flow rate at the exit of the second tank, h_1 *and* h_2 - the water heights in the first and the second tank, a_1 and a_2 - the opening of the leak valve (in mm^2).

The goal is to maintain the water level in the second tank at a desired value. The entering water flow rate q is provided by the pump P which is controlled by the voltage u. The water then flows from the first to the second tank. That flow (q_1) is a function of the difference between the water levels in the two tanks (h_1 and h_2). The water fow at the bottom of the second tank is a function of the water level in that tank (h_2). Therefore, the system input is the pump's voltage u and the system output is the water level in the second tank h_2).

The mathematical model consists of the following equations:

$$\begin{aligned} \frac{V_1(t)}{dt} &= q(t) - q_1(t), \\ \frac{V_2(t)}{dt} &= q_1(t) - q_2(t), \end{aligned} \tag{1}$$

where $V_1(t)$ and $V_2(t)$ are the volumes of the fluid contained in the first and second tank, respectively.

Both tanks are the same size and shape with identical dimensions, so the change of the water volume in the tank depends of the water level change:

$$\frac{V_i(t)}{dt} = A\frac{dh_i(t)}{dt}; i = 1,2. \tag{2}$$

The flow rate provided by the pump is a linear function of the pump's voltage:

$$q(t) = ku(t). \tag{3}$$

The water flow speed at the bottom of the tank is defined by the Torricelli's law:

$$v = \sqrt{2gh}, \tag{4}$$

where h is the height of the fluid, and $g = 9.81 m/s^2$.

The water flow through the pipes at the bottom of the tanks can be expressed in the following way:

$$\begin{aligned} q_1(t) &= a_1 v_1 = a_1 \operatorname{sgn}(h_1(t) - h_2(t))\sqrt{2g(h_1(t) - h_2(t)}, \\ q_2(t) &= a_2 v_2 = a_2\sqrt{2gh_2(t)}. \end{aligned} \tag{5}$$

The symbols q_1, h_1 and q_2, h_2 represent the flow rates and

978-1-4244-2904-2/08/$25.00 ©2008 IEEE

heights of the fluid in the first and second tank, respectively, a_2 is the opening of the valve connecting the first and second tank and a_2 is the opening of the second tank leak valve.

The complete nonlinear mathematical model is obtained by combining equations (5) and (1), thus resulting in:

$$A\dot{h}_1(t) = ku(t) - a_1 \operatorname{sgn}(h_1(t) - h_2(t))\sqrt{2g|(h_1(t) - h_2(t)|},$$
$$A\dot{h}_1(t) = a_1 \operatorname{sgn}(h_1(t) - h_2(t))\sqrt{2g|(h_1(t) - h_2(t)|} - a_2\sqrt{2gh_2(t)}.$$

III. THE NEURAL NETWORK CONTROLLERS

The NARMA-L2 neural control is sometimes referred to by another name: feedback linearization control. It is referred to as feedback linearization when the plant model has a particular form (companion form). The central idea of this type of control is to transform nonlinear system dynamics into linear dynamics by canceling the nonlinearities.

The first step in using the NARMA-L2 control is to identify the system to be controlled. You train a neural network to represent the forward dynamics of the system. One standard model that is used to represent general discrete-time nonlinear systems is the nonlinear autoregressive-moving average (NARMA) model:

$$y(k+d) =$$
$$N[y(k), y(k-1),..., y(k-n+1), u(k), u(k-1),..., u(k-n+1)]$$

where $u(k)$ is the system input, and $y(k)$ is the system output. For the identification phase, you could train a neural network to approximate the nonlinear function N. The controller used in this section is based on the NARMA-L2 approximate model:

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

This model is in companion form, where the next controller input $u(k)$ is not contained inside the nonlinearity. The advantage of this form is that you can solve for the control input that causes the system output to follow the reference $y(k+d) = y_r(k+d)$. The resulting controller would have the form

$$u(k) = \frac{y_r(k+d) - f[y(k), y(k-1),..., y(k-n+1), u(k-1),..., u(k-+1)]}{g[y(k), y(k-1),..., y(k-n+1), u(k-1),..., u(k-n+1)]}$$

Using this equation directly can cause realization problems, because you must determine the control input $u(k)$ based on the output at the same time, $y(k)$. It is better to use the model

$$y(k+d) =$$
$$f[y(k), y(k-1),..., y(k-n+1), u(k), u(k-1),..., u(k-n+1)]$$
$$+ g[y(k),..., y(k-n+1), u(k),..., u(k-n+1)] \cdot u(k+1)$$

where $d \le 2$. The following figures show the structures of the NARMA-L2 neural network, the NARMA-L2 controller as well as the NARMA-L2 controller and the process connected.

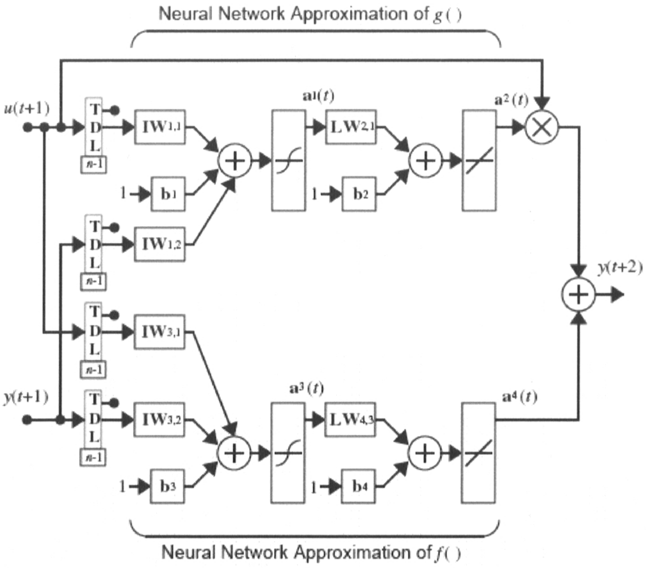

Fig. 2. The structure of the NARMA-L2 neural network

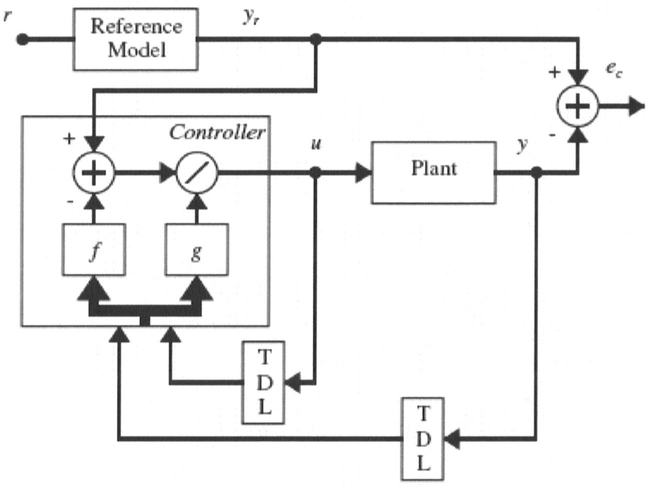

Fig. 3. A block diagram of the NARMA-L2 controller

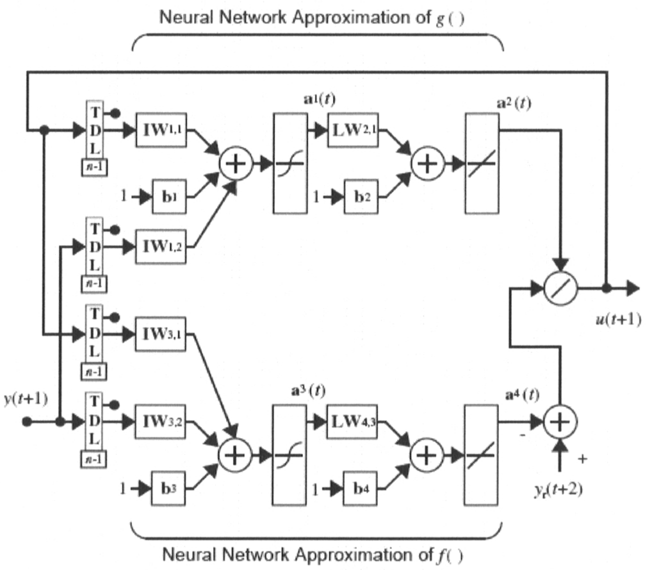

Fig. 4. Connecting the NARMA-L2 with the process

The data needed to train the neural network is gathered from a Matlab simulation of the process, with a sampling time $T = 5s$.

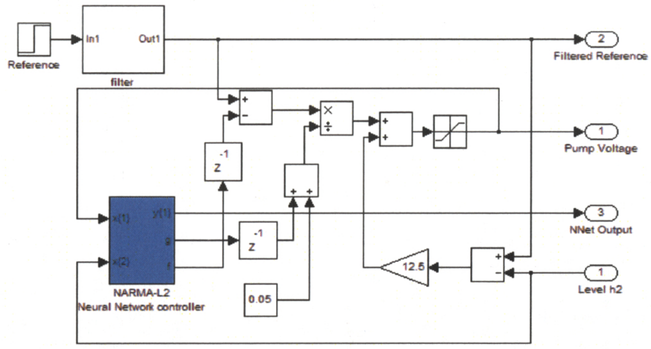

Fig. 5. Connecting the NARMA-L2 with the process in Simulink

IV. Results

The results of the conventional PID controller and the NARMA-L2 controller are shown on the diagrams below.

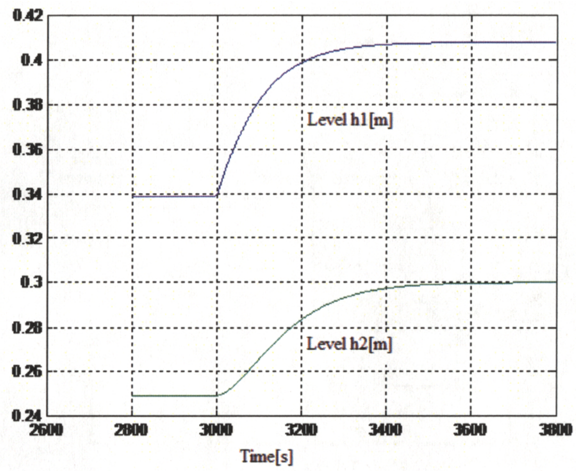

Fig.6. The process response when using a standard PID controller

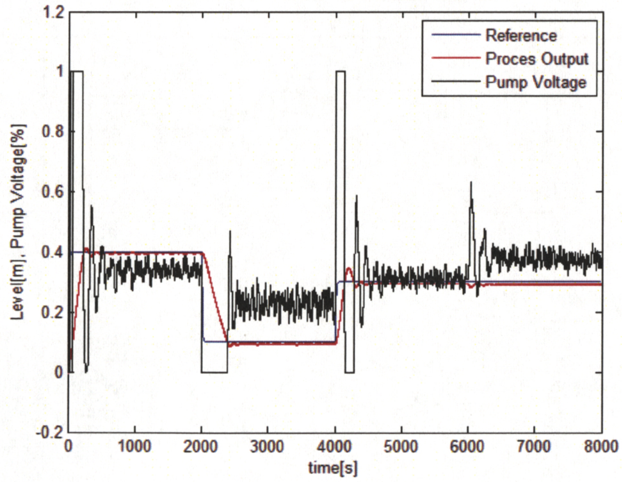

Fig. 7. NARMA-L2 process control; the refence changes at t=2000s and t=4000s; the disturbance occurs at t=6000s

We can clearly see that a significant increase of system response speed has been achieved. Inevitably, the overshoot had been increased (its value is around 15% of the steady state output value) as well as the steady state error, which is now around 2.5%. In order to perform a more thourough test of the controller, a step disturbance was simulated by openenig the leak valve of the second tank at t=6000s, changing its cross section from 13.5mm^2 to 15.4mm^2. The controller supressed the effects of the increased flow rate by increasing the voltage of the pump. The result was a slight augmentation of the steady state error.

The Model Reference Control architecture uses two neural networks: a controller network and a plant model network, as shown in the following figure. The plant model is identified first, and then the controller is trained so that the plant output follows the reference model output.

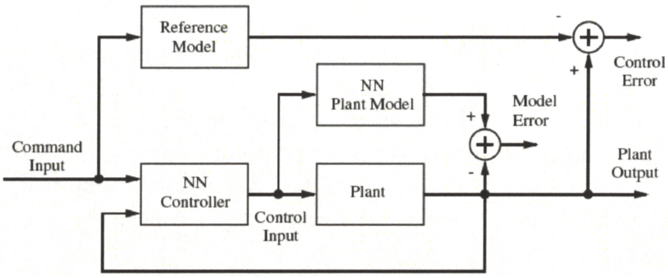

Fig. 8. The block diagram of the model reference control system

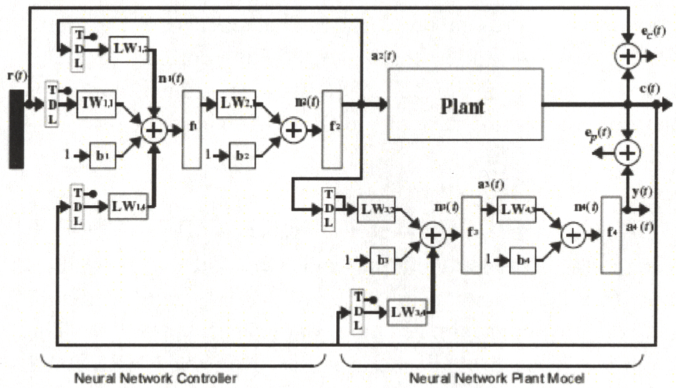

Fig. 9. Connecting the model reference control to the system

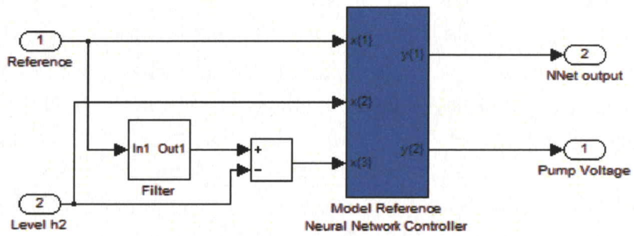

Fig. 10. Connecting the Model Reference Control to the system in Simulink

Each neural network has two layers. The detailed network structure as well as the connections between the network and the system are shown in the figure above. The data needed to train the neural networks is gathered from a Matlab simulation of the process, with a sampling time $T = 25s$.

The first step is is to train the neural network that 'imitates' the process. This can be done by constructing a separated two-layer neural network that uses training data which is generated using the Simulink process model.

As we can see from the diagrams, compared to the conventional PID controller, there has been a even greater improvement in system response speed in this case. Compared to the NARMA-L2 controller, the overshoot is increased (it is now around 30%) but the steady state error is reduced to a value very close to zero. The Model Reference Controller was exposed to the same disturbance as the NARMA-L2 controller and showed very similar results.

It is important to point out that all the neural networks were trained using the data generated from the nonlinear model, which is not the exact replica of the process itself, while the controllers were tested on the very process.

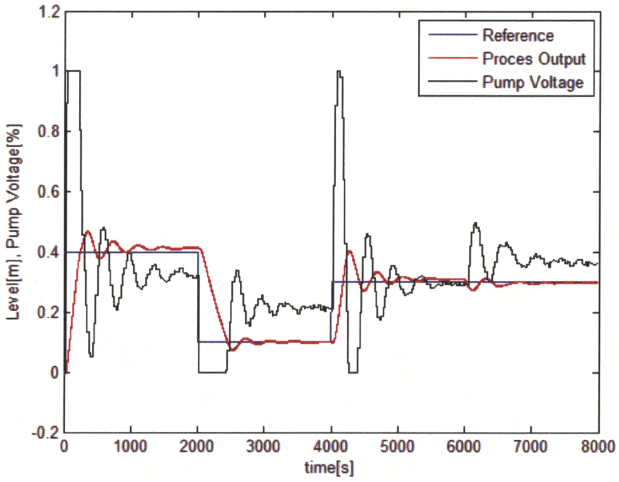

Fig. 11. The Model Referene Control; the refence changes at t=2000s and t=4000s; the disturbance occurs at t=6000s

V. Conclusion

Both NARMA-L2 neural controller and the Model Refence Control proved to be rather good controllers, considering that their application resulted in a significant decrease in process response time without heavilly increasing the response overshoot and the steady state error. Also an imporant fact: the PID controller, when tuned, can only be used in the working regime that it was designed for, while the neural network controllers, once trained, can be successfully applied to a wider spectrum of working regimes without any further tuning.

References

[1] *AMIRA DTS200 Laboratory Setup Three-Tank-System,* Copyright Amira GmbH, 2000,2002.

[2] *A Neuro-Fuzzy Synergism to Intelligent Systems,* Chin-Teng Lee and C. S. George Lee, Prentice Hall, 1996.

[3] *Neural Network Toolbox, For Use With Matlab®,* Howard Demuth, Mark Beale and Martin Hagan, The MathWorks, 2007.