

ELMAN NEURAL NETWORKS IN MODEL PREDICTIVE CONTROL

David Samek
Department of Production Engineering
Tomas Bata University in Zlin
Nam. TGM 5555, 76001 Zlin, Czech Republic
E-mail: samek@ft.utb.cz

KEYWORDS

Model predictive control, Elman neural network, predictor

ABSTRACT

The goal of this paper is to present interesting way how to model and predict nonlinear systems using recurrent neural network. This type of artificial neural networks is underestimated and marginalized. Nevertheless, it offers superior modelling features at reasonable computational costs. This contribution is focused on Elman Neural Network, two-layered recurrent neural network. The abilities of this network are presented in the nonlinear system control. The task of the controller is to control the liquid level in the second of two interconnected cylindrical tanks. The mathematical model of the real-time system was derived in order to test predictor and consequently the controller in Matlab/Simulink simulations.

INTRODUCTION

Model predictive control (MPC) (Camacho and Bordons 2007) is a very popular concept for the development and tuning of nonlinear controllers in the presence of input, output or state constraints. Many predictive control techniques based on MPC that use artificial neural network (ANN) as a predictor are established on multilayer feed-forward neural networks (Hagan et al. 2002), (Kanjilal 1995). In spite the multilayer feed-forward neural networks (MFFNNs) have many advantages such as simple design and scalability, they have also many drawbacks such as long training times and choice of an appropriate learning stop time (the over-learning versus the early stopping problem). However, there is quite a number of types ANNs suitable for the modelling and prediction, for instance adaptive linear networks, radial basis function networks and recurrent networks (Liu 2001), (Meszaros et al. 1999), (Koker 2006).

This paper is divided as follows: After short introduction to the recurrent neural networks, the used model predictive controller is explained. Then the model of the real time system is derived. After that the identification of the predictor (training of the artificial neural network) is described. When the identification is finished, the paper focuses on the model predictive

control and evaluation of results. The contribution is finished by some concluding remarks.

RECURRENT NEURAL NETWORKS

Recurrent neural networks (sometimes are these networks called feedback neural networks) can be distinguished from feed-forward neural networks in that they have a loopback connection (Figure 1). In its most general form recurrent network consist of a set of processing units, while the output of each unit is fed as input to all other units including the same unit. With each link connecting any two units, a weight is associated which determines the amount of output a unit feeds as input to the other unit (Yegnanarayana 2005).

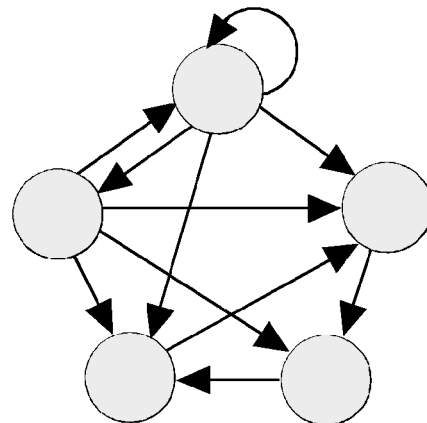


Figure 1: Example of Recurrent Neural Network

Recurrent neural networks have superior temporal and spatial behaviours, such as stable and unstable fixed points and limit cycles, and chaotic behaviours. These behaviours can be utilized to model certain cognitive functions, such as associative memory, unsupervised learning, self-organizing maps, and temporal reasoning (He 1999).

Elman Neural Networks

One of the most known recurrent neural networks is Elman neural network (Elman 1990). Typical Elman network has one hidden layer with delayed feedback. The Elman neural network is capable of providing the standard state-space representation for dynamic systems. This is the reason why this network architecture is utilized as a recurrent neural equalizer.

Generally, this network is considered as a special kind feed-forward network, including additional memory neurons and local feedback (Koker 2006). Typical structure of Elman neural network is depicted in fig. 2.

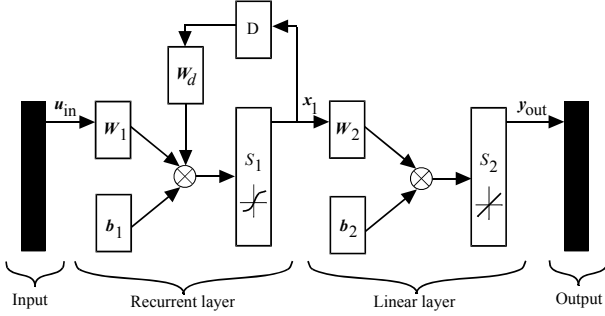


Figure 2: Elman Neural Network

MODEL PREDICTIVE CONTROL

There are various approaches to predictive control by artificial neural networks. Generally we can say that these methods use ANN as the plant model in order to get its output predictions. The most used approach is model predictive control (Camacho and Bordons 1995). MPC is a broad control strategy applicable to both linear and nonlinear processes.

The main idea of MPC algorithms is to use a dynamical model of process to predict the effect of future control actions on the output of the process. Hence, the controller calculates the control input that will optimize the performance criterion over a specified future time horizon:

$$J = \lambda \cdot \sum_{i=N_1}^{N_2} [w(k+i) - \hat{y}(k+i)]^2 + \rho \cdot \sum_{i=1}^{N_u} [u_t(k+i-1) - u_t(k+i-2)]^2 \quad (1)$$

where N_1 , N_2 and N_u define horizons over which the tracking error and the control increments are evaluated (usually $N_2 \geq N_u$). The u_t variable is the tentative control signal, w is the desired response and \hat{y} is the network model response. The parameters λ and ρ determine the contribution that the sums of the squares of the future control errors and control increments have on the performance index.

There is usually assumed that after a certain interval $N_u < N_2$ there is no variation in the proposed control signals, that is:

$$\Delta u(k+i) = 0 \quad \text{for } i \in \langle N_u, N_2 - 1 \rangle \quad (2)$$

This is equivalent to giving infinite weights to the changes in the control from a certain instant. This approach is adopted in this paper.

SIMULATIONS

In this chapter, the mathematical model of the real time system is derived first. Then the identification of the predictor is presented. After that the MPC controller is used in Matlab/Simulink for control of nonlinear system.

Modelling the Real-time System

Let us consider SISO nonlinear system to be controlled which is shown in the figure 3 and consist two connected cylindrical tanks for liquid. The dynamic model if the system is (considering usual simplifications):

$$q_{1v} = q_1 + \frac{\pi d_1^2}{4} \cdot \frac{dh_1}{dt} \quad (3)$$

$$q_1 = q_2 + \frac{\pi d_2^2}{4} \cdot \frac{dh_2}{dt} \quad (4)$$

$$q_1 = k_1 \sqrt{h_1 - h_2} \quad (5)$$

$$q_2 = k_2 \sqrt{h_2} \quad (6)$$

where d_j is the diameter, h_j is liquid level and q_j is the output flow of the j -th tank. q_{1v} is the input flow to the first tank. The constants associated with the properties of pipes and valves are k_1 and k_2 . The objective of the controller is to maintain the level in the second tank h_2 (output value) by adjusting the input flow q_{1v} (control signal).

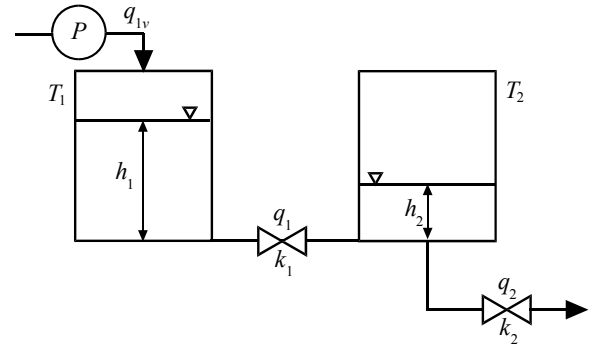


Figure 3: Scheme of Two Interconnected Cylindrical Tanks

This system is based on real-time laboratory model Amira DTS-200 (figure 4) which consist 3 cylindrical tanks and two pumps. However, in this contribution only two tanks (T_1 and T_2) and one pump were selected. Thus, the valves V2 and V4 were fully closed and the valve number 5 was set to the half position.

The values of parameters are $k_1=11.53 \text{ cm}^{2.5}/\text{s}$ and $k_2=13.09 \text{ cm}^{2.5}/\text{s}$ were obtained by experiments on the DTS-200. The maximum input flow q_{1v} is $100 \text{ cm}^3/\text{s}$. The height of the tanks is 60 cm and their diameter is 14 cm.

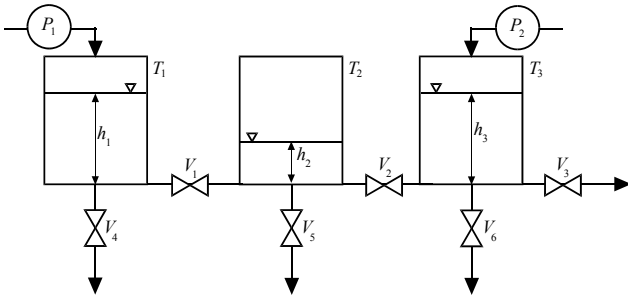


Figure 4: Scheme of Amira DTS-200

The mathematical model was transformed into the Simulink as is depicted in the figure 5. The Simulink model is more general and consist possibility to model TITO (two inputs – two outputs) system with two pumps and observation of level of liquid in both tanks.

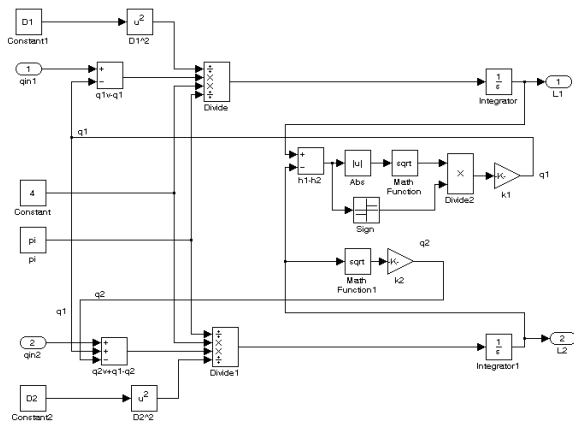


Figure 5: Simulink Model of the Two Cylindrical Tanks System

Predictor Based on Elman Neural Network

As was mentioned hereinbefore, in this paper Elman Neural Network was used as a predictor of the controlled system. The structure follows figure 2. The predictor consisted of five neurons with hyperbolic tangent transfer function in hidden layer and one neuron with linear transfer function in the output layer.

In this chapter two variants of predictor are introduced. The first one used five last values of control signal and output signal, thus it had ten inputs. This predictor is further referred to as *predictor1*. The second predictor utilized only five past values of the output signal. This predictor is further referred to as *predictor2*.

Both predictors were trained offline for identification data obtained using stepped actuating signal. For the off-line identification was used input-output data generated by pulses of random amplitude and duration. Duration and amplitude of the pulses must be chosen carefully to produce accurate identification. The amplitudes in range $<0; 100>$ cm³/s and duration from 1s to 50s were used. The Levenberg-Marquardt

backpropagation algorithm implemented in the Matlab Neural Network Toolbox was applied as training function. Final predictors were tested on the training data (figures 6 and 8) and unknown testing data as well (figures 7 and 9).

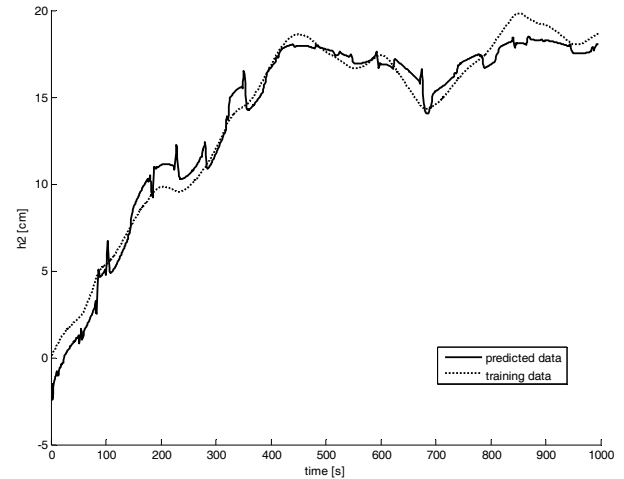


Figure 6: Training Data Response of *predictor1*

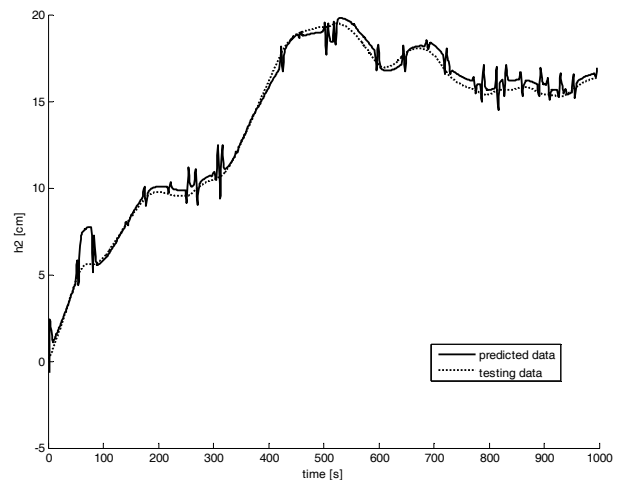


Figure 7: Testing Data Response of *predictor1*

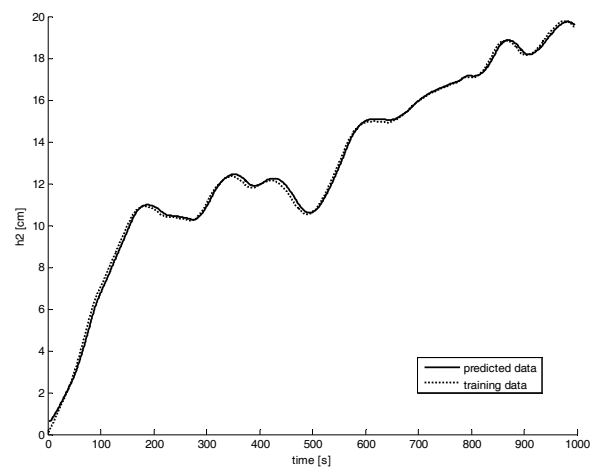


Figure 8: Training Data Response of *predictor2*

As can be seen from figures, *predictor1* suffers from a disturbance. This is caused by step-shaped identification control signal. Each of step bounce of predicted signal is caused by step change of control signal at the predictor input. This behaviour can be easily removed by omitting the control signal at the predictor input, as is illustrated in figures 8 and 9 (*predictor2*). Difference between both predictors is shown in the Table 1, where are presented mean square prediction error (MSPE) values for from 1 to 5-step ahead predictions of the same testing data.

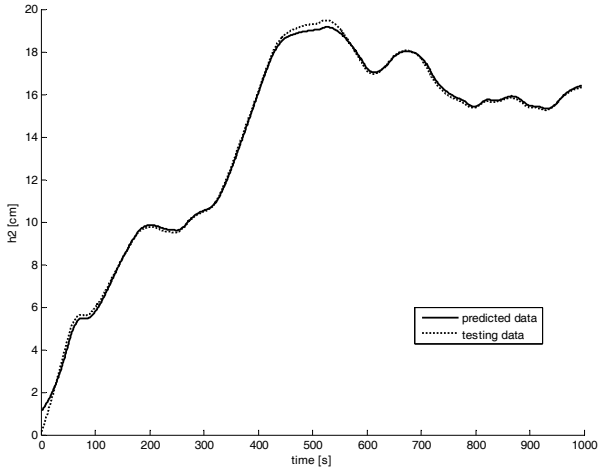


Figure 9: Testing Data Response of *predictor2*

Table 1: Mean Square Prediction Errors – Comparison of the Predictors

	MSPE				
	1 step	2 step	3 step	4 step	5 step
<i>predictor1</i>	2,79	5,89	9,11	13,51	18,66
<i>predictor2</i>	0,86	1,72	2,81	4,92	6,95

Control of Two Connected Cylindrical Tanks

Simulations were done for both predictors. As can be seen from figure 10 the control loop was simulated in Matlab/Simulink. The sampling period of Simulink was set to 1s. Because the SISO system is regarded, the input flow to the second tank is set to zero and the level of liquid in the first tank is not used.

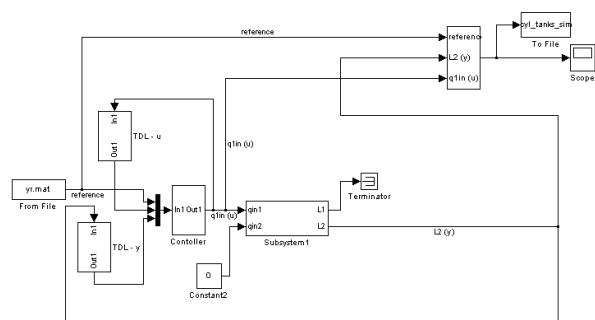


Figure 10: Simulink Scheme of the Control Loop

Due to constraints and nonlinear nature of predictors numerical optimization of the MPC criterion was necessary. The controller used constrained quasi-Newton method from Matlab Optimization Toolbox as a nonlinear optimization algorithm.

TDL blocks in the figure 10 represents so called Tapped Delay Line which stores five recent values of the signals. This block is necessary because artificial neural network uses five recent values of system input and output. Structure of TDL for is depicted in figure 11.

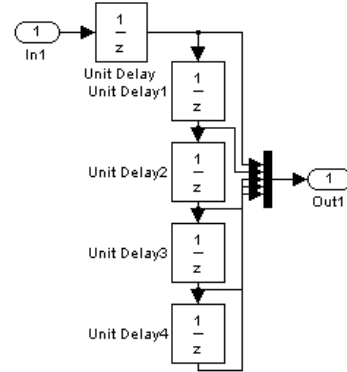


Figure 11: TDL Block from the Simulink Control Loop

A vast number of simulations were done due to find optimal setting of the controller. From comparability reasons the same controller settings is used for the both predictors: $\lambda = 1$, $\rho = 0.001$, $N_1 = 1$, $N_2 = 5$, $N_u = 5$. Simulations results are presented in figures 12 and 13.

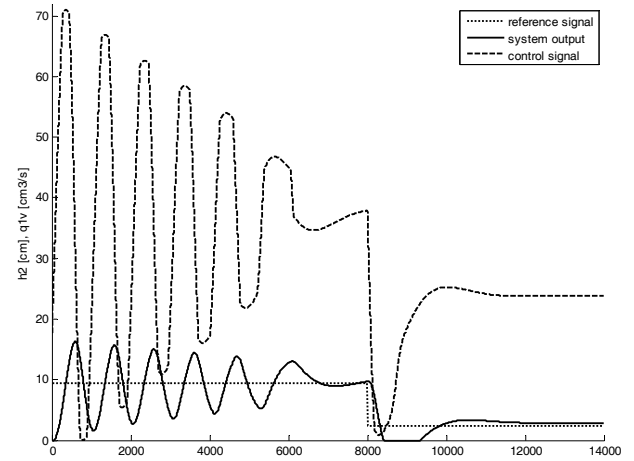


Figure 12: Control using *predictor1*

In order to compare results of both controllers (predictors) we used two quadratic criterions:

$$S_y = \sum_k (w_k - y_k)^2 \quad (7)$$

$$S_u = \sum_k (u_k - u_{k-1})^2 \quad (8)$$

The first criterion S_y is based on control errors and represents the tracking performance of the controller. While the second criterion S_u is based on changes of the control signal and represents the controller demands on the actuators. The computed values of both criterions are shown in the table 2.

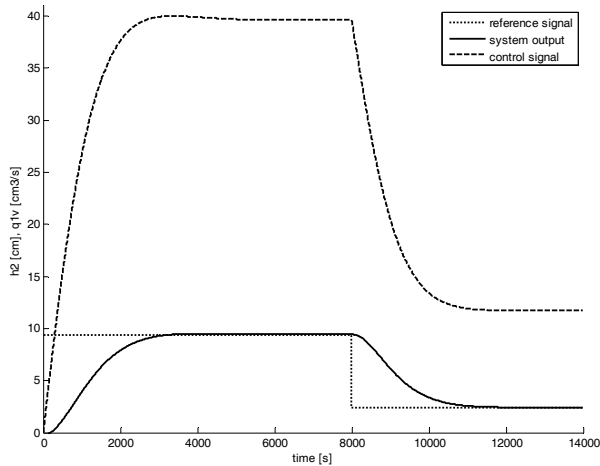


Figure 13: Control using *predictor2*

Table 2: Comparison of the Controllers/Predictors Using the Quadratic Criterions

Predictor Used in the Controller	Quadratic Criterions	
	S_y [cm ²]	S_u [cm ⁴ /s ²]
<i>predictor1</i>	$1,27 \cdot 10^5$	112,08
<i>predictor2</i>	$1,18 \cdot 10^5$	1,31

CONCLUSIONS

As can be seen simulations, the usage of Elman neural networks as a predictor in model predictive control is possible, while special attention should be paid to modelling stage. The quality of predictor and its predictions is influenced by its structure and identification data.

The badly identified predictor (*predictor1*) didn't perform well in prediction task. However, it is worth noticing that model predictive controller is quite tolerant to a predictor inaccuracy, because (as can be seen from figure 12) the system output still tracks the reference.

The most significant aspect of this paper is the reduction in computational time. The model predictive control method involves highly mathematical computations. What is more, predictors based on artificial neural networks significantly increase computational demands of the MPC controllers. Nevertheless, Elman neural network provides very interesting way how to reduce

computational costs, because the training times of Elman networks are incredibly short. This kind of artificial neural network could be promising for on-line adaption of the predictor in case of t-variant systems. This will be subject of further research.

ACKNOWLEDGEMENTS

This work has been supported by the Ministry of Education, Youth and Sports of the Czech Republic under grant MSM 7088352102 and by the Grant Agency of the Czech Republic under grant 102/07/P137. This support is greatly acknowledged.

REFERENCES

- Camacho, E. F. and C. Bordons. 2007. *Model Predictive Control – Second edition*. Springer – Verlag, London.
- Elman, J.L., 1990 "Finding structure in time," *Cognitive Science*, Vol. 14, 179-211.
- Hagan, M.; Demuth H.; and O.D. Jesus. 2002. "An Introduction to the Use of Neural Networks in Control Systems". *International Journal of Robust and Nonlinear Control*, No.12 (Nov), 959-985.
- He, Q. 1999. *Neural Network and Its Application in IR*. University of Illinois, Urbana-Champaign
- Kanjilal, P.P. 1995. *Adaptive prediction and predictive control*. P. Perengrinus Ltd., London.
- Koker, R. 2006. "Design and performance of an intelligent predictive controller for a six-degree-of-freedom robot using the Elman network." *Information Sciences*, Vol. 176, Issue 12, (Jun), 1781-1799.
- Liu, G.P. 2001. *Nonlinear identification and control – A neural network approach*. Springer - Verlag London Ltd., London.
- Meszáros, A.; Rusnak, A.; and M. Fikar. 1999. "Adaptive Neural PID Control – Case Study: Tubular Chemical Reactor," *Computers and Chemical Engineering Supplement*, No.1999, 847-850.
- Yegnanarayana, B. 2005. *Artificial Neural Networks*. PHI Learning Pvt. Ltd., New Delhi.

AUTHOR BIOGRAPHIES

DAVID SAMEK was born in Zlin, Czech Republic and went to the Brno University of Technology, where he studied automation and control and obtained his degree in 2000. Then he studied post gradual in Tomas Bata University in Zlin and obtained Ph.D. degree in 2005. Since 2001 he has been working at Tomas Bata University in Zlin as a senior lecturer. His field of interest is focused on artificial networks and their applications. His e-mail address is : samek@ft.utb.cz and his Web-page can be found at <http://www.ft.utb.cz/czech/uvi/>

