An Effective Application of Soft Computing Methods for Hydraulic Process Control

Ladislav Körösi and Štefan Kozák

Institute of Control and Industrial Informatics,
Faculty of Electrical Engineering and Information Technology,
Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava, Slovak Republic
{ladislav.korosi,stefan.kozak}@stuba.sk

Abstract. The article deals with the modeling and predictive control of real hydraulic system using artificial neural network (ANN). For the design of optimal neural network model structure we developed procedures for creation optimal-minimal structure which ensure desired model accuracy. This procedure was designed using genetic algorithms (GA) in Matlab-Simulink. The predictive control algorithm was implemented using CompactLogix programmable logic controller (PLC). The main aim of the proposed paper is design of methodology and effective real-time algorithm for possible applications in industry.

Keywords: Neural network, genetic algorithm, predictive control, PLC realization, optimization methods.

1 Introduction

Neural networks are currently used in various fields such as signal processing, image recognition, natural speech recognition, identification and others [1]. Functions in programmable logic controllers (PLC) libraries are simple (bit operations, summation, subtraction, multiplication, division, reminder after division, etc.) or complex (sine, cosine, absolute value, vector summation, etc.) mathematical functions but without artificial neural systems, while PLC systems are currently the most commonly used control systems in industry. In PLC systems are also missing matrix operations and often vector operations, generally said parallel mathematical operations. The proposed paper has the objective to demonstrate the real deployment of optimal neural network for modeling and predictive control of hydraulic system.

2 Realization of Neural Network Control Algorithm by Programmable Logic Controllers

Modeling and Control of nonlinear processes using artificial neural networks in practice can be solved in two ways. The first way is the deployment of modeling and control algorithm in the master system (SCADA, Application running on the PC, local HMI, etc.). In this case, the algorithm is separated from the control system and therefore it is important to ensure trouble-free communication between these parts of control as well as fixed sampling time. In the event of a failure of communication,

V. Snasel et al. (Eds.): SOCO Models in Industrial & Environmental Appl., AISC 188, pp. 379–388.
springerlink.com

must take control the local control system. The second solution is the implementation of intelligent algorithms directly to the control system. Most commonly used control systems are programmable logic controllers (PLC's). PLC is a digital computer used for automation of electromechanical processes, such as control of machinery on factory assembly lines, etc. PLC's are used in many industries and machines. For the programming of such control systems have been introduced IEC 61131-3, which unifies the programming languages of PLC's from different manufacturers. The best language for the purposes of the implementation of the ANN appears is Structured Text (ST), which is similar to the Pascal programming language. Of course there are differences in them because this language was developed for PLC programming. The program is written in free style, making it clearer and more readable. It is particularly useful for the expression of different data types, structures and complex mathematical calculations [10]. Most PLC's do not support matrix operations or dynamic allocation of one-dimensional or multidimensional vectors, therefore, the implementation phase of the ANN learning and processing can be difficult. Variables must be pre-allocated, i.e. in the case of smaller ANN structures consumes unnecessarily PLC memory. Matrix operations can be programmed accordingly, but in this case they are not parallel operations, but sequential, decreasing the computing power of ANN [8]. That does not mean that affects the quality of modeling and control. Implementation of optimal ANN structures in the PLC is possible in two ways. The first way is to implement the general algorithm, which can dynamically adapt to new structures depending on given parameters. The second is the fixed structure of the ANN implemented in the PLC.

3 Artificial Neural Network Model

Perceptron is the most used ANN. The main reason is its ability to model as simple as well as very complex functional relations. Kolmogorov's theorem says that the perceptron ANN with one hidden layer and a sufficient number of neurons in this layer can approximate any nonlinear function. In Fig. 1 shows an example of the structure of three-layer perceptron.

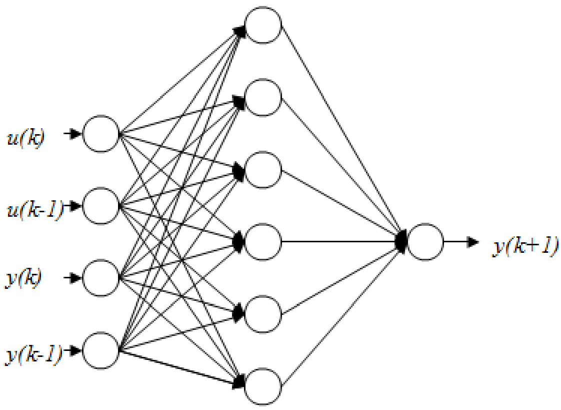

Fig. 1. An *example of multilayer perceptron structure*

The input and output neurons are linear activation functions (AF) but in some cases, in the output layer(s) can be used non-linear activation functions. Most often used activation functions in the hidden layer(s) are nonlinear (sigmoidal). Sigmoidal function is defined as a monotonically increasing, smooth and bounded function. Sigmoid activation functions allow obtaining high neuron sensitivity for small signals, for higher signal level the sensitivity decreases. Neural network is able to process signals in a sufficient range of dynamics without threatening to overload with too large coming signals. In addition to these activation functions provide nonlinear behavior of ANN and are several times differentiable, therefore they can be used for several techniques for ANN training and predictive control [2], [3], [5].

4 Artificial Neural Network Structure Optimization

Selection of optimal artificial neural network structure includes the determination of number of hidden layers, number of neurons in each layer, number of links between neurons, etc. which can be in general defined as the vastness of the network. ANN structure optimization methods can be divided into the following groups: construction algorithms, destruction algorithms, empiric methods, combined methods, genetic algorithms, etc. Optimal ANN usually contains fewer neurons and connections allowing their usage in real-time applications [7], [9]. Genetic algorithm is a universal stochastic search approach inspired by both natural selection and natural genetics which is able to approximate the optimal solution within bounded solution space. The method is capable being applied to a wide range of problems including ANN structure optimization. Basic objects in the GA are chromosome, gene, population, generation and fitness function [4].

Currently there are various methods for neural network structure encoding. One of the most used and simple method is encode NN structure with direct encoding. For the direct encoding of the connections between neurons binary *1* (*true*) and *0* (*false*) are used. If neurons are interconnected, the connection is assigned with value *1*. If the neurons aren't interconnected, the connection is assigned with value *0*. For all such links we can define the size of the *NxN* matrix that defines connections between all neurons. Each line defines the link neurons to neurons in a given row. From such a matrix is then created a chromosome for the GA. This encoding enables to optimize all interconnections between neurons.

Finding the optimal structures of ANN for modeling and control of nonlinear processes using GA is time consuming, especially for large ranges in chromosome genes. In addition, the GA has to evaluate the fitness function and decode the structure of ANN as well as check the correctness and optimality of newly established structures. Based on the fact that a three-layer ANN is sufficient for nonlinear process modeling is proposed the following ANN with AF encoding [7]:

1. gene: Number of input neurons for the control signal in history
2. gene: Number of input neurons for the measured signal in history
3. gene: Number of hidden neurons
4. gene: AF type

The sum of the number of input neurons for history of control and measured values defines the total number of input neurons. Number of output neurons is given by the process type and isn't encoded in the chromosome. All genes are encoded with integer number (also for AF type). During decoding the structure of ANN is the activation function of each number assigned to its name from the list of defined activation functions. Table 1 gives several examples of representation of the gene for AF types. Chromosome above defines the NNARX (neural network auto-regressive model with external input) structure (Fig. 1).

Table 1. AF type encoding

Gene: AF type	
Num. value	**AF type**
1	**tansig**
2	**logsig**
3	**modified tansig**

5 Predictive Control Using ANN

Predictive control methods currently represent a large group of modern control methods with an increasing number of applications. Under the notion of predictive control we understood a class of control methods where the mathematical model is used to predict the future output of the controlled system. Determination of the future control sequence involves minimizing a suitable criteria function with predicted increment of control and deviation. One of the advantages of predictive control is the possibility to use any process model [6]. In this article an ANN perceptron is used as process model.

Standard criteria function includes square of the deviation and control increment:

$$J_r = \frac{1}{2}\alpha(u(k) - u(k-1))^2 + \frac{1}{2}[r(k+1) - y_M(k+1)]^2 \tag{1}$$

where y_m is the output of the ANN.

The optimization block calculates the control signal so that the predicted output of the ANN matches the process output [4]. This is an iteration process which has the form:

$$u(k)_{new} = u(k)_{old} - \beta \frac{\partial J_r'}{\partial u(k)_{old}} \tag{2}$$

6 Case Study

For the verification and testing of proposed algorithm for modeling and control we consider real hydraulic system. The block diagram of the hydraulic system is depicted in Fig. 3.

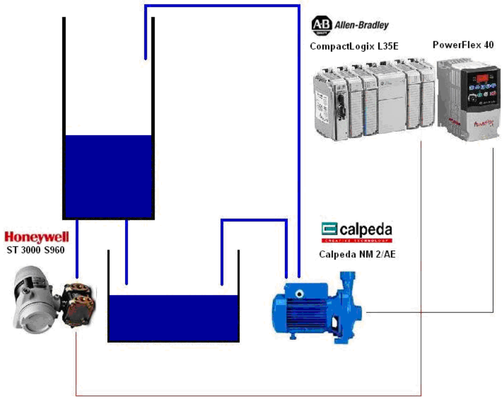

Fig. 2. Block *diagram of the hydraulic system*

Short description of used components is listed below.

CompactLogix (1769-L32E) - Programmable logic controller from Allen-Bradley is used to control the hydraulic system's level. The level is measured with pressure transmitter connected to analog input module (using unified 4-20mA signal). The control signal from analog output module is connected via 4-20mA to the inverter sending 0-100% (0-50Hz) signal.

PowerFlex 40 - Inverter from Allen-Bradley is designed for drives with power output from 0.4 kW to 11kW. It is connected to pump which pumps water from buffer situated under the tank to the upper part of the tank.

ST 3000 S900 - A smart pressure transmitter from Honeywell is used to measure the water level in tank with free drainage.

Calpeda NM 2/AE 400V 0.75kW - Monoblock centrifugal pump from the Calpeda company

Measurement of input-output data for identification of the real system can be implemented in two ways. The first way is to backup data using visualization (RSView32 or RSView Studio), but this method does not guarantee accurate sampling period - loading data from the PLC. The second way is to backup data directly in the PLC and their export using the Tag Upload Download after measurement. In our case we used a second approach for data collection to identify and record values during the process control. Measured data (level - depending on the pressure sensing and control variable) are shown in Fig. 4 and Fig. 5.

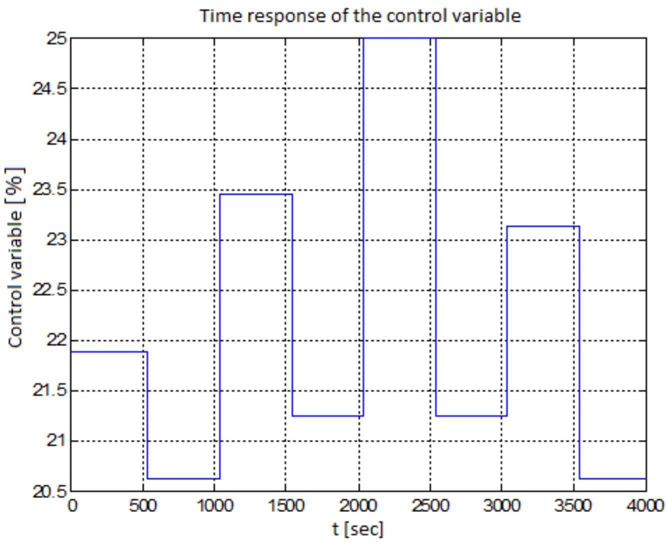

Fig. 3. Time *response of the control variable*

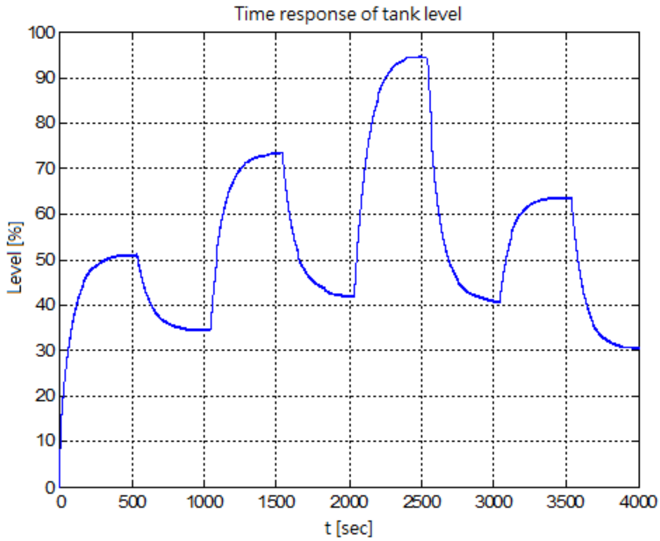

Fig. 4. Time *response of controlled variable (level)*

The setup of the GA was the following:

- 1. Gene value interval <1,6>
- 2. Gene value interval <1,6>
- 3. Gene value interval <1,30>
- 4. Gene value interval <1,3>
- Number of generations: 100 (used as stop criteria)
- Number chromosomes in population: 15

The criterion function (fitness) was calculated by the value of the square sum of deviations between ANN output and the process output.

For each ANN training new weights were generated. The found optimal neural network structure for modeling of real physical system was 4-6-1 with tansig AF. Inputs to the ANN were values $u(k)$, $u(k-1)$, $y(k)$, $y(k-1)$ and the output value was the predicted tank level $y(k+1)$. Structure of an ANN was created in Matlab and trained with back-propagation learning method with over-learning testing. Comparison of time responses of the real plant level and ANN output is shown in Fig. 7.

Structure of the ANN control algorithm for real time control was implemented in RSLogix5000 for CompactLogix PLC. Weights and biases of ANN have been exported from Matlab to RSLogix5000. The chosen control algorithm was gradient-descent. The results for different parameters (affecting different control quality) are shown in the following figures (for sampling time 100ms).

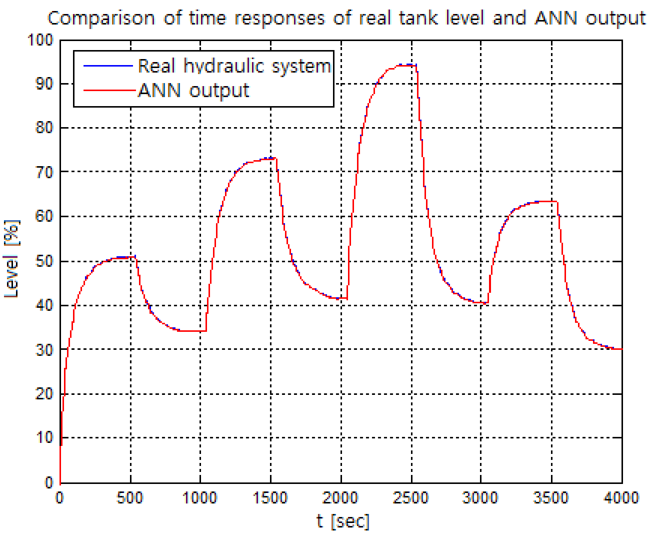

Fig. 5. Comparison *of time responses*

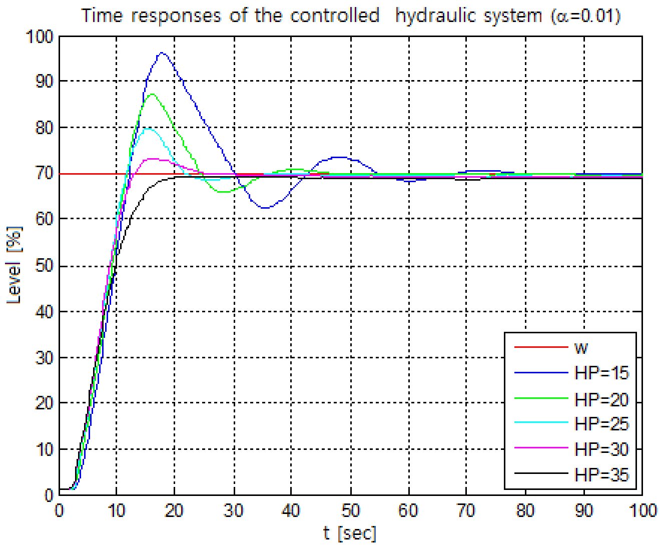

Fig. 6. Time responses of the controlled hydraulic system for different prediction horizons 15, 20, 25, 30 and 35 (α=0.01)

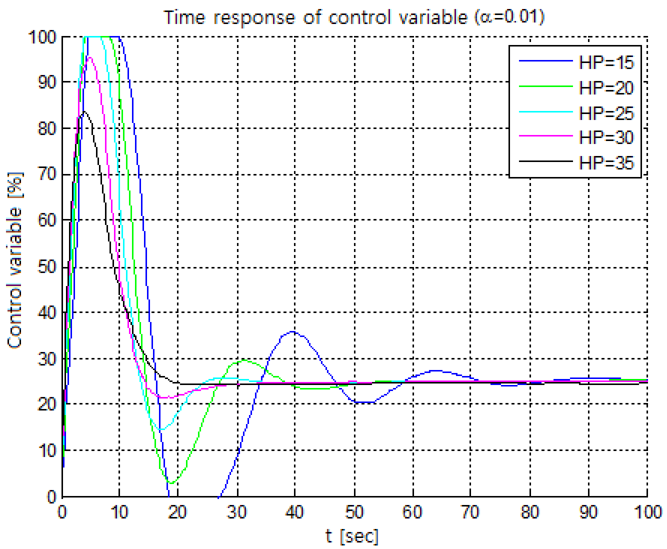

Fig. 7. Time responses of the control variable for different prediction horizons 15, 20, 25, 30 and 35 (α=0.01)

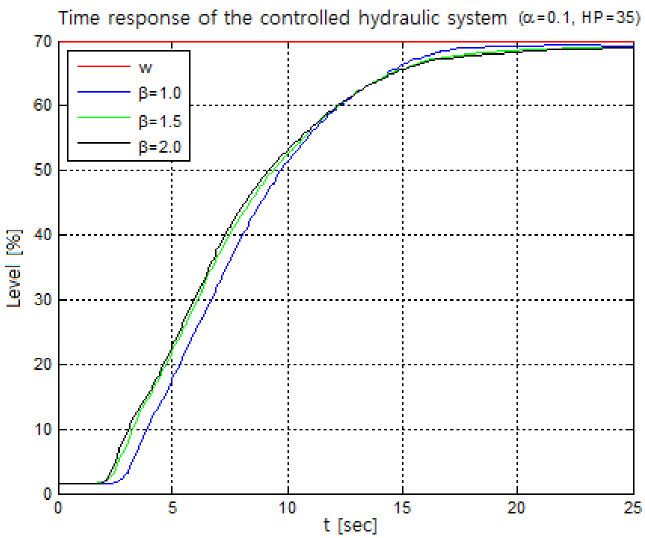

Fig. 8. Comparison of time responses of the controlled hydraulic system for different β and constant prediction horizon 35 and $\alpha=0.1$

7 Conclusion

Intelligent control belongs to the class of control techniques like artificial neural networks, fuzzy logic, evolutionary algorithms, etc. The performance of the proposed methodology was verified on several simulation examples. In the proposed paper was presented a practical application using optimized artificial neural network structure to identify and control the hydraulic system. Using these proposed methods we achieved better approximation results and faster response. The advantages of the proposed approach are minimal (optimal) neural network structure and faster signal processing therefore it's suitable for real-time control in industry control systems.

Acknowledgement. This paper was supported by the Slovak Scientific Grant Agency VEGA, Grant no.1/1105/11.

References

1. Oravec, M., Polec, J., Marchevský, S.: Neurónové siete pre číslicové spracovanie signálov. Vydavateľstvo FABER, Bratislava (1998) (in Czech)
2. Haykin, S.: Neural networks – A Comprehensive Foundation. Macmillan College Publishing Company, New York (1994)
3. Sinčák, P., Andrejková, G.: Neurónové siete – Inžiniersky prístup I, Technická univerzita, Košice (1998), `http://www.ai-cit.sk/source/publications/books/NS1/html/index.html` (in Slovak)

4. Körösi, L., Kozák, Š.: Optimal Self Tuning Neural Network Controller Design. In: 16th IFAC World Congress, Praha, Czech Repubic (2005)
5. Jadlovská, A.: Modelovanie a riadenie dynamických procesov s využtím neurónových sietí, Košice (2003) ISBN 80-88941 -22 -9 (in Slovak)
6. Dideková, Zuzana – Kajan, Slavomír: Riadenie nelineárnych systémov pomocou neurónových sietí optimalizovaných genetickými algoritmami. In: Kybernetika a informatika: Medzinárodná konferencia SSKI SAV, Vyšná Boca, SR, 10.-13.2, Vydavateľstvo STU, Bratislava (2010) ISBN 978-80-227-3241-3 (in Slovak)
7. Körösi, L.: Príspevok k problémom optimalizácie štruktúr umelých neurónových sietí. Dizertačná práca. Ústav riadenia a priemyselnej informatiky FEI STU, Bratislava, 126s (2010) (in Slovak)
8. Abdi, H., Salami, A., Ahmadi, A.: Implementation of a new neural network function block to programmable logic controllers library function. World Academy of Science, Engineering and Technology 29 (2007)
9. Kajan, S.: Modelovanie a riadenie systémov pomocou neurónových sietí s ortogonálnymi funkciami v prostredí MATLAB. In: Technical Computing Prague (2006): 14th Annual Conference Proceedings (in Slovak)
10. Mrafko, L., Mrosko, M., Körösi, L.: PLC a ich programovanie 1. Čo je to PLC? In: Posterus (2010) ISSN 1338-0087, `http://www.posterus.sk/?p=6903` (in Slovak)