

# System Identification in Difficult Operating Conditions Using Artificial Neural Networks

Petar Matić, Ivana Golub Medvešek, Tina Perić

To investigate an ability of system identification in difficult operating conditions. A simulation based experiment was performed on a simple second order system with white noise signal superimposed to the output signal. Interferences are added to the output signal in order to simulate difficult operating conditions present in a real system environment. Based on system simulation measurements, the system was identified using conventional method with least squares estimate and an alternative method, a multi-layer perceptron (MLP) network. Graphical evaluation of simulation results showed that MLP network produced better results than conventional model, with significantly better results in case of interferences in the output signal. To model dynamic system, a simple two-layer perceptron network with external dynamic members was trained in Matlab using Levenberg-Marquardt algorithm.

## KEY WORDS

- ~ System identification
- ~ Difficult operating conditions
- ~ Artificial neural network
- ~ Multy-layer perceptron
- ~ Levenberg-Marquardt.

## 1. INTRODUCTION

The model of the system is often used in research to perform experiments instead of a real system, to optimize system performances and to design system control. Basic controller design principle says that controller is possible to design only if the system and its environment are known (Astrom and Wittenmark, 1989), meaning a system is possible to model. Model reference adaptive control systems (MRAC), discussed in (Astrom and Wittenmark, 1989) and (Narendra and Parthasarathy, 1989), predictive control systems, discussed in (Astrom and Wittenmark, 1989) and (Al Seyab and Cao, 2008), as well as methods of controller settings, such as self-tuning regulator (STR) described in (Astrom and Wittenmark, 1989), are all based a model of the system. Therefore, the possibility of forming a reliable system model is of great importance.

However, each system model represents only a fairly good approximation of the actual system, which means that the model as such will inevitably differ from the actual system. Generally, model can be formed based on the mathematical formulation of the physical processes that occur in the system being modelled and that model is called physical model. The model can also be formed on the basis of the measured values of input and output variables of the system. That model is called the experimental model, and the process of forming the model is called identification, i.e. parameter identification. The experimental model, although it does not give insight to the physical properties of the system, is easier to form and better describes the input-output behaviour of the system. This property makes it suitable for control system designing, as well as prediction of the system behaviour. The identification can be performed

during the operation of the system (on-line identification) or through specially prepared experiment (off-line identification), as described in a third chapter of this paper.

It is possible to use alternative methods for system identification, such as artificial neural networks (ANNs) briefly described in the second chapter of this paper. Over the years ANNs have emerged as an important tool for representing nonlinear transformations and have proved successful in system identification and control. To identify the system using ANN means to optimize network parameters to achieve the minimum objective function, i.e. to train the network. As can be found described in detail in (Petrović, 1998), a system identification of a non-linear dynamic processes can be performed with a static neural network. A static neural network is preferably used over a dynamic network primarily because of its structural stability and ease of training. In order to use static neural network for dynamic system modelling, external dynamic members are added to the network. Where, it is possible to use a different model structure, such as NARX structure used in this paper and described in section 3.2.

System identification using ANN does not represent a new approach, since it has been used to model different systems, as described in (Seyab and Cao, 2008), (Amlashi et al., 2013), (Narendra and Parthasarathy, 1989), (Leyva et al., 1997) and (Asgari et al., 2013). Furthermore, it has been shown that neural network model is more accurate in comparison to conventional nonlinear models, as in the example from paper (Amlashi et al., 2013). However, the quality of ANN model is largely influenced by the quality of a measurements performed on a system, and sometimes the interferences are inevitable. The goal of this paper is to investigate the possibility of identifying a system in difficult operating conditions that are reflected in a noisy system output signal. The basic idea is to make use of the ANN's generalisation ability to improve the model identification in case of difficult operating conditions, since network is expected to adapt to unknown operation conditions better than a conventional model. Difficult operating conditions are common in all types of plants, but especially ships where measured values are affected by different disturbances such as vibration, waves, temperature, etc. Therefore, it is expected that models identified using artificial neural networks could mean an improvement in system modelling for the case of difficult operating condition.

## 2. ARTIFICIAL NEURAL NETWORKS

Inspired by biological neural networks, artificial neural networks (ANNs) represent a simplified mathematical model of the process that occurs in a brain of the living beings. ANN resembles the brain in two respects. First, knowledge is acquired by the network through a learning process, and second, interneuron connection strengths, known as synaptic weights, are used to

store the knowledge (Haykin, 1998). Therefore, ANN is an artificial structure that consists of a number of interconnected artificial neurons and resembles to the brain in terms of information processing and storage. Based on a type of a neuron and the way they are connected different ANNs have been developed. Essentially, they all operate in a similar manner. Using a value of input variables ( $x_{n,t}$ ) and network parameters (synaptic weights and biases,  $w_{n,t}$ ) at instant  $t$ , every neuron calculates its response ( $y_t$ ) based on the expressions (1) and (2), where  $\phi$  represents an activation function, and  $p_t$  is a sum of weighted inputs. Model of an artificial neuron defined by (1) and (2) can be shown in "Figure 1" for better understanding.

$$p_t = \sum_{n=0}^m (w_n \cdot x_t) \quad (1)$$

$$y_t = \phi(p_t) \quad (2)$$

ANNs have a number of properties amongst which: learning from example, generalization and nonlinearity represent the most distinguished ones. During the process of learning, a number of input-output pairs are presented to the network as examples of the system behaviour. Based on the known inputs, network output is calculated and, and since the desired output is also available, the error, i.e. the difference between the network output and the desired output, can also be calculated. Learning algorithm then uses the error information to adjust network parameters ( $w$ ) in order to reduce the error ( $e$ ). The second most important property of ANN, the generalization, can be interpreted as the network ability to extract the general rule from the examples it has been exposed to during training, and solve examples it has not encountered during the training process. And finally, nonlinearity, i.e. ability to approximate nonlinear functions, is defined by the properties of neuron activation function, for which sigmoidal functions are often used.

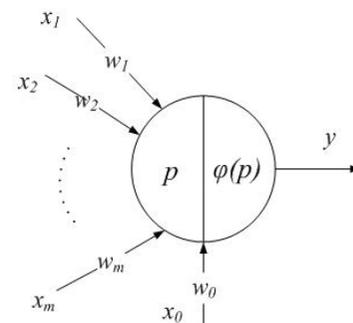


Figure 1.  
Artificial neuron.

When referring to ANNs as a tool, it may be said that ANNs represent a data driven technique for modelling systems based on available data. This enables modelling of a system where it is possible to define input and output variables and measure them. In the initial moment, the network has no information on how the system works nor does it know the relationship between input and output variables. It introduces a system through a process of learning (i.e. training). Once trained, ANN can be used to generate the value of an output variable based on the values of input variables, i.e. to simulate system behaviour.

### 2.1. Multi-Layer Perceptron (MLP)

As different neural networks exist in nature, different ANN types have been formed to model to a different properties of a system. However, static feed-forward ANN called Multi-Layer Perceptron (MLP) represents the most commonly used network for modelling processes. Figure 2 displays a structure of an MLP neural model, also known as universal approximator with  $m$  inputs,  $h$  hidden neurons and  $o$  output neurons. Adjustable network parameters (i.e. network weights,  $w$ ) have been marked with respect to the neurons they are connecting, using double indexes; and based on the corresponding layer, using the exponent; as shown in Figure 2, thus using a three-dimensional notation. However, for the training purposes they are marked using a one-dimensional notation, with a single index. The relation between these notations is following:  $w_1 = w^1_{1,1}$ ;  $w_2 = w^1_{1,2}$ ; ...;  $w_N = w^2_{o,o}$ . MLP network with universal approximation properties uses nonlinear sigmoidal activation functions in a hidden layer and linear activation function at the output layer neurons. Although optimal number of layers was often subject of research, it has been proven that two-layer structure is sufficient to approximate any practical function, given enough neurons in hidden layer (Cybenko, 1989). Therefore, a two-layer perceptron network is used in this paper for system identification purposes.

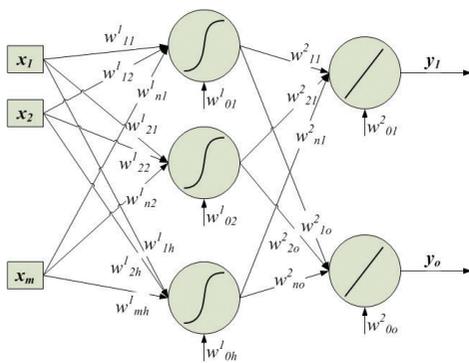


Figure 2. A Two-layer perceptron network.

### 2.2. Training of MLP Network

Although defined by its structure, i.e. number and type of neurons, ANN takes a final form only after learning process is over. The procedure used to perform a learning process is called learning (or training) algorithm. Its function is to modify the network parameters in an orderly fashion to attain a desired objective, i.e. to minimize error between network output and desired value. A basic algorithm developed for training MLP network is error backpropagation (EBP), which could be regarded as one of the most significant breakthroughs in field of neural networks. Although EBP algorithm is still widely used, many improvements have been made to the original algorithm to improve its slow convergence.

In this paper, Levenberg-Marquardt (LM) algorithm is used to train MLP network. LM algorithm is a second order local algorithm for training feedforward ANNs, which can be briefly described in four steps, as stated below, or explained in more detail in (Hagan and Menhaj, 1994) and (Hao Yu and Wilamowski, 2011). First, the network is excited and the output ( $y$ ) is calculated based on expressions (1) and (2), for the set of current input values ( $x$ ) and network parameter values ( $w$ ). Sum of square error (SSE), defined by the expression (3), is then used to get information about total error for the training set of input-output pairs. The output error ( $e_{s,o}$ ), defined by (4), is calculated based on desired network output ( $d$ ) and actual network output ( $y$ ) for every pair  $s$  and every output ( $o$ ).

$$SSE = \frac{1}{2} \sum_{s=1}^{N_s} \sum_{o=1}^{N_o} e^2_{s,o} \quad (3)$$

$$e_{s,o} = d_{s,o} - y_{s,o} \quad (4)$$

Secondly, a Jacobian matrix which is calculated according to the expression (5), where  $N$  is a total number of network parameters,  $N_s$  is number of input-output pairs and  $N_o$  is the output number.

$$J = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w_1} & \frac{\partial e_{1,1}}{\partial w_2} & \dots & \frac{\partial e_{1,1}}{\partial w_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_{s,K}}{\partial w_1} & \frac{\partial e_{s,K}}{\partial w_2} & \dots & \frac{\partial e_{s,K}}{\partial w_N} \end{bmatrix} \quad (5)$$

Thirdly, new values of the adjustable network parameters ( $w_{t+1}$ ) are calculated according to the expression (6) and SSE is calculated again. Finally (step 4), if the new value of the SSE is higher than the one calculated in step 1, the parameter  $\mu$  is multiplied by a factor of constant amount ( $\kappa$ ). If the step in the training process results in a reduction of SSE, the parameter  $\mu$  is divided by factor  $\kappa$ . Steps 1 to 4 are repeated until the objective function (SSE) minimum is achieved, or until progression of error reduction becomes negligibly small.

$$w_{t+1} = w_t - (J_t^T \cdot J_t + \mu \cdot I)^{-1} J_t^T \cdot e_t \quad (6)$$

A key problem in LM algorithm is to determine the elements of the Jacobian matrix. The computation process for Jacobian matrix can be organized according to the traditional backpropagation computation. Therefore, elements of the Jacobian matrix can be calculated based on the expression (7), where  $\delta_j$  stands for the local gradient and  $x_j$  stands for an input of neuron  $j$ . Since an input to the second layer neurons is equal to the output from the first layer ( $y_{j-1} = x_j$ ), value of  $x_j$  is calculated in the forward computation, while  $\delta$  is obtained in the backward computation, and it depends on the position of neuron  $j$ . For the output layer neurons  $\delta$  can be calculated based on (8), and for hidden neurons it is calculated based on (9).

$$\frac{\partial e_{s,k}}{\partial w_{ji}} = -\delta_j \cdot x_j \quad (7)$$

$$\delta_j = e_j \cdot \varphi'_j(p_j) \quad (8)$$

$$\delta_j = \varphi'_j(p_j) \cdot \sum_1 (\delta_{j+1} \cdot w_{ji}) \quad (9)$$

### 3. SYSTEM IDENTIFICATION

System identification is model determination process based on the measured values of the input and output signals of the system. As discussed in (Petrović, 1998), identification process can generally be divided into five basic steps, which are: input-output data acquisition, model structure determination, evaluation measure selection, i.e. quality criteria, model parameter estimation, validation of the model.

If data is collected during normal operation, the process is called on-line identification, as it is the case with the adaptive systems. If the input-output data is collected during specially prepared

experiment, then the process is called off-line identification. For the identification experiment purposes a Band Limited White Noise signal (BLWNS) is used to excite the system. The identification experiment is usually performed in open loop, except in cases where, due to the instability of the system or of production reasons, a closed loop is used. A key step of the system identification process is a structure determination. A flexible structure that can describe a large class of different systems is usually used. This structure represents so-called black box model since it does not model in detail exact physical processes that occur in the system. A second order transfer function is usually used based on the assumption that most physical systems could be approximated using a second order system. Selected structure has unknown parameters that can be determined by one of the available methods, and in this paper least squares estimate (LSE) is used. If the model satisfies the validation process, it is accepted as a valid model of the system, and if not, the identification process is repeated. Validation is carried out using the excitation signal with a different random numbers than the ones used during identification.

In addition to conventional identification methods, there are alternative methods, such as the application of artificial neural networks. ANNs ability to approximate nonlinear functions enables their use in the identification of nonlinear dynamical processes as described in section 3.2 of this paper.

#### 3.1. Least Squares Estimate (LSE) Method

If the structure of the discrete-time transfer function is used, as represented by the expression (10), total number of model parameters ( $\theta$ ) is than defined as  $L = 2m$ .

$$G_E(z) = \frac{\Theta_0 + \Theta_1 \cdot z^{-1} + \dots + \Theta_m \cdot z^{-m}}{1 + \Theta_{m+1} \cdot z^{-m+1} + \dots + \Theta_{2m} \cdot z^{-2m}} \quad (10)$$

LSE method for model parameters identification is based on using the same excitation for calculating model response  $\hat{Y}(k)$  and the actual system response  $Y(k)$  in order to determine the error of the model  $E(k)$ , as expressed with (11), where the upper case letters were used to represent the matrixes.

$$E(k) = Y(k) - \hat{Y}(k) \quad (11)$$

The excitation used represents a sequence of the system input  $u(k)$  and output  $y(k)$  signals, and can be presented by matrix  $H(k)$ , as defined by the expression (12). Identification goal is to estimate model parameters ( $\theta$ ) that would produce minimal SSE calculated for different time instants ( $k$ ).

$$H(k) = \begin{bmatrix} u(k) & u(k-1) & \dots & u(k-m) & -y(k-1) & \dots & y(k-m) \\ u(k-1) & u(k-2) & \dots & u(k-m-1) & -y(k-2) & \dots & y(k-m-1) \\ \vdots & \vdots & & & & & \\ u(k-L) & u(k-1-L) & \dots & u(k-m-L) & -y(k-1-L) & \dots & y(k-m-L) \end{bmatrix} \quad (12)$$

Mathematically, this condition can be expressed as (13), where  $W(k)$  is a weight matrix (usually diagonal, identity matrix), indicating the importance of discrete time instant used for calculation of the parameters of the impulse transfer function.

$$J(\theta) = E(k)^T \cdot W(k) \cdot E(k) \quad (13)$$

As explained in (Stipaničev and Marasović), equation (13) has a minimum value for the parameter  $\theta$  calculated from (14). Equation (13) gives the required values of the model parameters, which are then introduced into the expression (10) to obtain the model of the system in the form of a discrete transfer function.

$$\theta = \begin{bmatrix} [H^T(k) \cdot W(k) \cdot H(k)]^{-1} \\ [H^T(k) \cdot W(k) \cdot Y(k)] \end{bmatrix} \quad (14)$$

### 3.2. System Identification Using Artificial Neural Network

It makes perfect sense to use dynamic neural network for dynamic system modelling. However, dynamic neural networks are structurally unstable due to internal feedback connections. These feedback connections are making the process of learning more difficult, and also make stability analysis of the model necessary.

A dynamic system modelling using static ANN is possible if external dynamic members (time delay units,  $z^{-1}$ ) are added to the network. By adding external dynamic members, a regression vector is formed which represents the input vector for the static ANN. As described in (Petrović, 1998), different signals can be used to form regression vector. The most common structure used for non-linear system modelling is NARX structure, shown in "Figure 3", which uses the input ( $u$ ) and the output ( $y$ ) system signals to form the regression vector.

NARX model can be considered as a general model of non-linear systems. Its output is the estimated value of the output signal of the process, calculated on the basis of the currently available input and output signals of the process, according to the expression (15), where  $\varphi(k)$  stands for regression vector defined by the expression (16).

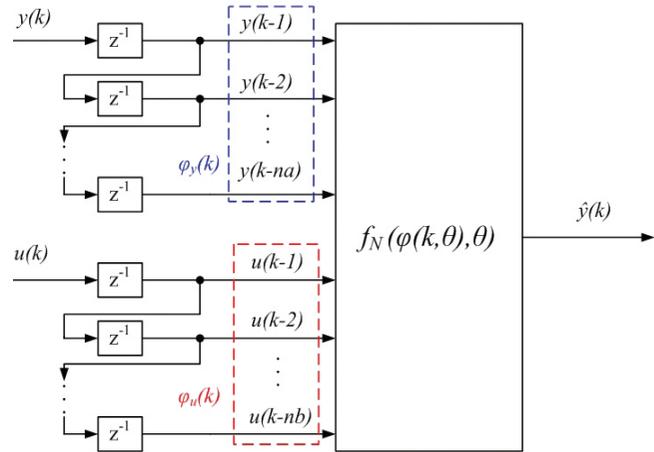


Figure 3.  
NARX model.

$$\hat{y}(k) = f_N(\varphi(k), \theta) \quad (15)$$

$$j(k) = [j_y(k), j_u(k)] = [y(k-1), \dots, y(k-na), u(k-1), \dots, u(k-nb)]^T \quad (16)$$

Therefore, a general model of nonlinear dynamical systems can be represented as a function composition. First function relates the space of the measured values of input and output signals to the regression space and second relates the regression space to the system output space. A static ANN can be used to model the second function, thus making the identification problem equal to the function approximation. The task of the parameter estimation using ANNs is to find the optimal values of the adjustable network parameters for which the error is minimal. This process is called learning or training and is described in section 2.2 of this paper.

#### 4. CASE STUDY: IDENTIFICATION OF A SECOND ORDER SYSTEM

A second order system defined with the expression (17) is used in this paper to investigate possibility of system identification using LSE method and ANN for the case of difficult operation conditions. For the simulation purposes, system parameters are chosen as  $K = a = 1$ . Values of the input and output signals of the system are recorded during simulation performed in Matlab/Simulink with a calculation step size  $T_s = 0.1$  s.

$$G(s) = \frac{K}{s(s+a)} \quad (17)$$

The white noise signal is superimposed to the system output to simulate the inevitable appearance of interference that affects the output of the system under real operating conditions in complex systems like ships. The input signal used to excite the system in order to gather the information for identification purposes represents a series of random numbers obtained as the sum of the white noise signal of limited frequency range (BLWNS) and the triangular signal.

##### 4.1. Second Order System Identification Using LSE Method

Structure selection in this case is facilitated since the actual structure of the system is known. Therefore, assumed structure is identical to the structure chosen for the simulation purposes (17). To proceed with identification, a Z-transformation is applied in order to determine discrete transfer function, according to the expression (18), and the equivalent discrete transfer function (19) is obtained.

$$G_E(z) = \frac{z-1}{z} \cdot Z\left[\frac{G(s)}{s}\right] \quad (18)$$

$$G_E(z) = \frac{\Theta_0 + \Theta_1 \cdot z^{-1} + \Theta_2 \cdot z^{-2}}{1 + \Theta_3 \cdot z^{-1} + \Theta_4 \cdot z^{-2}} \quad (19)$$

Given that the largest index of unknown parameters is  $L = 4$ , with arbitrarily chosen moment for performing identification (discrete instant  $k = 100$ ), discrete transfer function of a system can be identified using LSE method, as described in section 3.1 of this paper. In case of no interference in the output signal, the solution to the problem is presented by (20), while in case of interferences the solution is presented with the expression (21).

$$G_E(z) = \frac{0.004836 \cdot z^{-1} + 0.004675 \cdot z^{-2}}{1 - 1.90531 \cdot z^{-1} + 0.905214 \cdot z^{-2}} \quad (20)$$

$$G_E(z) = \frac{-0.0074 - 0.0420 \cdot z^{-1} + 0.0529 \cdot z^{-2}}{1 - 0.4275 \cdot z^{-1} - 0.5948 \cdot z^{-2}} \quad (21)$$

##### 4.2. Second Order System Identification Using Static MLP Network

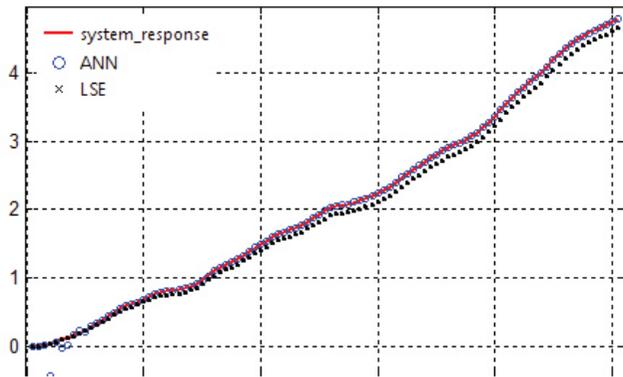
In order to perform system identification using static ANN, a regression vector is formed having 6 members, three successive excitation signals  $u(k-1)$ ,  $u(k-2)$ ,  $u(k-3)$  and three successive output signals  $y(k-1)$ ,  $y(k-2)$ ,  $y(k-3)$  of the system being modelled. While regression vector represents a network input, current values of the model output  $tt$  are used to form "target" vector for network training purposes. A double-layer perceptron is used, with first layer having 20 neurons and one neuron in the second layer. A neural network used has a structure of universal approximator which means that first (hidden) layer neurons have non-linear (tansig) and the output neuron uses linear (purelin) activation function. Although optimal number of layers was often subject of research, it has been proven in (Cybenko, 1989) that two-layer structure is sufficient to approximate any practical function, given enough neurons in hidden layer. To determine the initial values of network parameters Nguyen-Widrow function (initnw) was used. The network was trained with Levenberg-Marquardt algorithm, with mean square error (MSE) used as a performance function. The process of forming a neural model in Matlab can be found explained in (Hagan et al., 1994). Optimal network type, number of neurons, training algorithm, number and type of inputs and performance function were not investigated in this paper. Therefore, experiments and/or methods to determine optimal values of these parameters should be performed in order to achieve optimal results.

#### 5. SYSTEM IDENTIFICATION RESULTS

To examine the identification process results, models responses are compared as presented in figures 4 for case with no interference, and in figure 5 for the case with an interference in the output signal. During this test, an original system and the identified models were excited with a signal different from the one used for identification purposes.

As it can be seen from "Figure 4", both identification methods produce fair results, with ANN model response better fitting the actual system response. The LSE method is not an iterative

process and system identification is performed in an arbitrary step with a finite number of pre-recorded samples of input-output parameters. In order to reduce deviation response of LSE model a recursive least squares estimate (RLSE) could be performed.



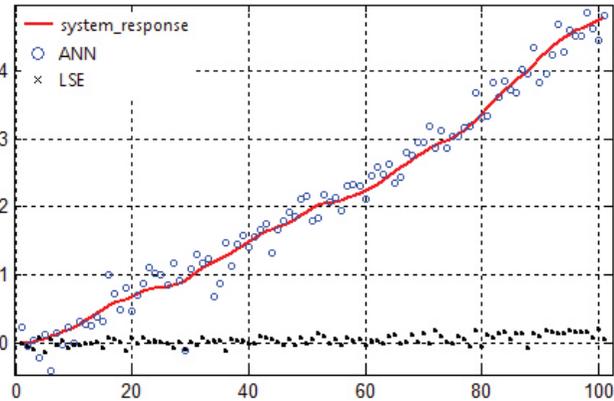
**Figure 4.**  
Model comparison for the case without interference in the output signal.

From the results shown in “Figure 5”, it is obvious that ANN is able to identify the system even in case when there are interferences present in the output signal, while LSE obtained model has no such ability. Neural model follows the response of the actual system, while LSE method model was not able to track the response of the actual system.

## 6. CONCLUSION

In this paper system identification was performed using conventional parameter estimation with least squares estimate (LSE) and alternative solution, identification with artificial neural network (ANN). The complexity of the system itself was not an issue; therefore a simple second order system was used as a case study, but rather the complexity of the environment. In order to simulate difficult operating conditions, i.e. complex system environment, white noise signal was superimposed to the output signal of the system being modelled. Interference in the output signal was used to simulate realistic operating conditions where noise usually occurs in the signal due to the industrial operating conditions and/or measurements imperfections. The main goal of the paper was to investigate the premise that successful identification could be performed in case of difficult operating conditions if artificial neural network was used. Therefore, two methods (LSE and ANN) were used to perform a system identification of a simple second order system in case of normal and difficult operating conditions. Both methods proved to be good solutions for the purpose of system identification in case of

no interference present in the output signal, with ANN obtaining a slightly better results.



**Figure 5.**  
Model comparison for the case with interference in the output signal.

However, in case of interference present in the output signal, LSE method could not produce satisfactory results, while ANN proved to be quite efficient. Based on the simulation results it can be concluded that the ANN justified the reputation of a flexible, easy-to-use tool for modelling systems even in difficult operating conditions. Results from this paper could use as a guidelines for further investigation which should be oriented towards ship's system identification and control design. For that purpose, an effort of adequate data acquisition should be made as prerequisite for the system identification.

## REFERENCES

- Al Seyab, R.K. and Cao Y., (2008), Nonlinear system identification for predictive control using continuous time recurrent neural networks and automatic differentiation, *Journal of Process Control*, 18(6), pp. 568-581., <http://dx.doi.org/10.1016/j.jprocont.2007.10.012>
- Amlashi, N.J.S., Shahsavari, A., Vahidifar, A. and Nasirian, M., (2013), Nonlinear System Identification of Laboratory Heat Exchanger Using Artificial Neural Network Model, *International Journal of Electrical and Computer Engineering (IJECE)*, 3(1), pp. 118-128.
- Asgari, H., Chen, X., Menhaj, M.B. and Sainudiin, R., (2013), Artificial Neural Network Based System Identification for a Single-Shaft Gas Turbine, *Journal of Engineering for Gas Turbines and Power* 135(9), pp. 092601-092607., <http://dx.doi.org/10.1115/1.4024735>
- Astrom, K.J. and Wittenmark, B., (1989), *Adaptive Control*, Reading, MA: Addison-Wesley.
- Beale, M.H., Hagan, M.T., Demuth, H.B., (2015), *Neural Network Toolbox™, User's Guide*, Natick, MA: Mathworks.

Cybenko, G., (1989), Approximation by Superpositions of a Sigmoidal Function, *Mathematics of Control, Signals and Systems* 2(4), pp. 303-314., <http://dx.doi.org/10.1007/BF02551274>

Hagan, M.T. and Menhaj, M.B., (1994), Training Feedforward Networks with the Marquardt Algorithm, *IEEE Transactions on Neural Networks*, 5(6), pp. 989 – 993., <http://dx.doi.org/10.1109/72.329697>

Haykin, S., (1998), *Neural Networks: A Comprehensive Foundation*, Upper Saddle River, NJ: Prentice Hall PTR.

Leyva, R., Martinez-Salamero, L., Jammes, B., Marpinard, J. C. and Guinjoan, F., (1997), Identification and Control of Power Converter by Means of Neural Networks, *IEEE Transactions on Circuits and Systems – I: Fundamental Theory and Applications*, 44(8), pp. 735 – 742., <http://dx.doi.org/10.1109/81.611270>

Narendra, K.S. and Parthasarathy, K., (1989), Adaptive Identification and Control of Dynamical Systems using Neural Networks, *Proc. of the 28th Conference on Decision and Control*, Tampa, Florida. December 13 – 15, 2: pp. 1737 – 1738., <http://dx.doi.org/10.1109/CDC.1989.70448>

Petrović, I., (1998), Identifikacija nelinearnih dinamičkih procesa statičkim neuronskim mrežama, doktorska disertacija, Zagreb: University of Zagreb, Faculty of Electrical Engineering and Computing.

Stipaničev, D. and Marasović, J., (2008), *Digitalno vođenje*, Split: university of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture.

Yu, H. and Wilamowski, B.M., (2011), Levenberg – Marquardt Training Industrial Electronics Handbook, vol. 5 – Intelligent Systems, 2nd Edition, chapter 12, pp. 12-1 to 12-16, Boca Raton: CRC Press.