

# DIAGNOSES OF ADDITIVE FAULTS OF SERIAL WOUNDED MOTOR USING ARTIFICIAL INTELLIGENCE METHODS

<sup>1</sup>Viktor FÜVESI, Associate research scientist, <sup>2</sup>Dr. Ernő KOVÁCS, PhD., A/Professor

<sup>1</sup>Department of Research Instrumentation and Informatics, Research Institute of Applied Earth Science, University of Miskolc, Miskolc, Hungary;

<sup>2</sup>Department of Electrical and Electronic Engineering, University of Miskolc, Miskolc, Hungary

[fuvesi@afki.hu](mailto:fuvesi@afki.hu), [elkke@uni-miskolc.hu](mailto:elkke@uni-miskolc.hu)

**KEYWORDS:** artificial intelligence, fault diagnostic, actuator

## Abstract:

In this paper an artificial neural network based technique will be introduced, which is capable to separate the different types of faulty state of the analysed system and generate signs to alarm the user about the failures in the system. The method can detect, separate, and identify the faults in the system. Large datasets were developed to train the separator networks. To speed up the training process of separator network, active learning method was applied. To find the separator mathematical structure weakness, a complex test process was used where the size of the different faults was varied and the performance of the structure was examined. The examination has two parts: first the appearance and termination of the faults were tested; later the estimation of the fault size was checked. The separator technique needs mathematical models of the analysed system. In our case, the models were also based on feed forward neural networks.

## 1. INTRODUCTION

Nowadays the complex systems such as industrial equipment or common systems must have the feature like accuracy, safety and reliability. State-of-art mathematical methods from hard and soft computing area are used to detect, separate and identification (Zhou, 2003) the different faults of industrial systems (Azhar, 2010) or dynamic systems (Nicholson, 1992). Several papers can be found which use Kalman filter (Spina, 2000), parity equations (Pranatyasto, 2001), principal component analyses (Zhang, 2003) for fault detection and diagnosis. Other authors use wide spectrum of different artificial intelligence methods like: Bayes networks (Mehranbod, 2005), probabilistic neural networks (Jabbari, 2007), multilayer perceptron (MLP) (Zhang, 2006) or fuzzy logic (Amann, 2001).

Also some example can be found in fault diagnostics of electro-mechanical actuator system. A DC motor second order state variable model was developed in (Padmakumar, 2009). The model was used to detect the fault of current measurement sensor using four different kinds of Kalman filters.

Fault detection and diagnoses of induction machine were discussed in (Sainz Palmero, 2005) using neuro-fuzzy based method. Several faults were separated which can be dated from the drive of the motor or sensors of the system.

MLP structure is used for fault detection of a simulated induction machine (Kolla, 2000).

Large portion of the published papers do not detail the process of evaluation of the fault separator structure, just the results are presented.

In this paper MLP based fault separator structure will be introduced which able to separate four additive faults of an actuator system and identify the fault size too. The method is multi-model based method. The required models will be also presented. To analyze the performance of the algorithm and compare the different structured separator networks a complex test process will be demonstrated.

Several network structures were trained. To reduce the computation time of the training process, an active training algorithm was used with the basic Rprop algorithm. The pros of algorithm will be shortly brought forward.

## 2. DATA ACQUISITION ON THE REAL SYSTEM

The analysed system, the actuator was a serial wounded DC motor. This motor type is typically used in vehicles to start the combustion engine during the start process. The engine needs a base rotation of speed to satisfy continuous operation in short period of time.

A test rig (Figure 1.) was design to measure the main physical parameters of the motor. Wide range of load torque characteristics was applied to the motor to analyse the dynamics of the system. An externally controlled powder brake was connected to the shaft of the motor as a dynamic load.

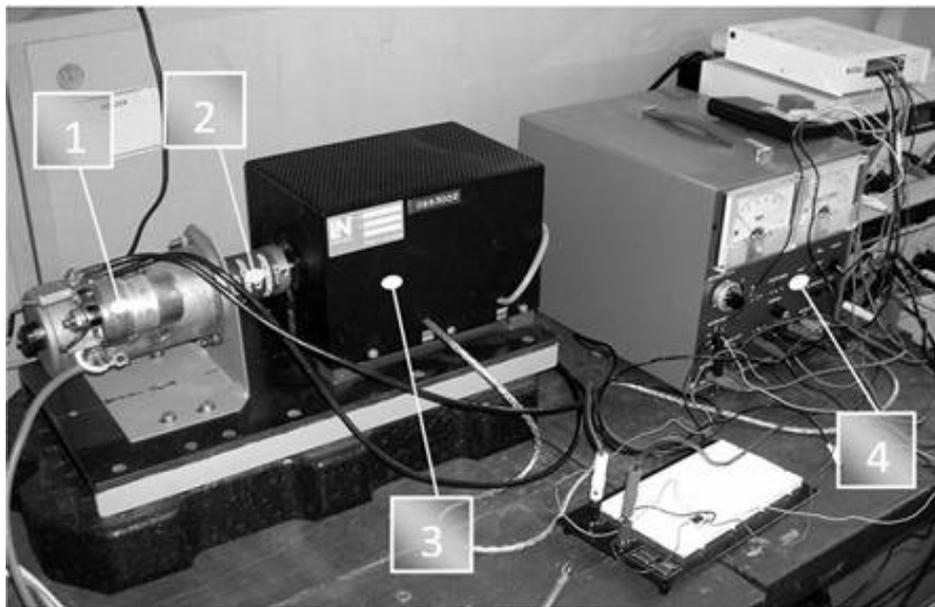


Fig. : Laboratory measurement rig

1-starter motor; 2-clutch, 3-powder brake; 4-controller of powder brake

The measured electrical parameters were: excitation current, voltage of the accumulator, which supplied the power of the motor. The brake includes velocity and torque transducers, so the angular velocity and the torque of the shaft of the motor were measured parallel with the previous parameters. Range of the measured parameters can be seen in Table 1.

Table 1. Range of different measured parameters

Parameters	lower range	upper range
Armature current [A]	0	300
Voltage of the battery [V]	9.5	13
Velocity of the motor [rpm]	0	5000
Load torque [Nm]	0	5

The control and measurement system was built up with National Instruments based equipment, the control and measurement software was written in Windows CVI. This system was able to carry out the data acquisition meanwhile control the load simultaneously. The sampling frequency was set to 800 Hz.

### 3. DESIGNED OBSERVERS

In generally differential equation based (Ciurys, 2009) or state space models can be used to describe the behaviour of the actuator system. The development process of these methods needs to include two steps: model development and identification. The friction nonlinearities, thermal effects, measurement noises can make it difficult to identify the parameters of the models so sometimes easier to use some soft computing method. One opportunity to use black box models like neural networks. Using the network structure single (SISO) (Füvesi, 2011) or multi input single output (MISO) models can be developed.

Using the result of the measurements two different multilayer feedforward neural network (NN) based observers were designed. One model was able to estimate the velocity and another was used to estimate the torque of the motor with good accuracy. The developed models used the measure current and voltage of the battery as inputs, so the developed models belong to the family of MISO systems.

To reach a good result several parameters of NN were varied and the results compared. During the simulations the different network structures were analysed. The two main structures were the NNFIR (Eq. 1) and NNARX (Eq. 2) NN models (Norgaard, 2000). These models make one step ahead prediction using the data of past of system.

$$o(t) = f [i(t-1), i(t-1-d), i(t-1-2d), \dots, i(t-1-nd), u(t-1), u(t-1-d), u(t-1-2d), \dots, u(t-1-nd)] \quad (1)$$

$$o(t) = f [i(t-1), i(t-1-d), i(t-1-2d), \dots, i(t-1-nd), u(t-1), u(t-1-d), u(t-1-2d), \dots, u(t-1-nd), y(t-1)] \quad (2)$$

where  $o(t)$  - is the estimated output of the network in the  $t^{\text{th}}$  time step, it can be velocity or torque signal;  $i(t-1)$ ,  $u(t-1)$  and  $y(t-1)$  - are in order the current, voltage signals and required output of the system in  $(t-1)^{\text{th}}$  time step;  $d$  - is the size of time delay between the used samples.  $n$  - is number of used time delays.

To find the best model, several configurations of the developed neural networks were analyzed. The  $d$  was varied from 10 to 220 in 10 size step;  $n$  was changed from 1 to 4.

The chosen activation function of the neurons in both hidden and output layer were Elliott function which in many cases give as good result as sigmoid function but the time of computation is less (Füvesi, 2011). In generally the size of hidden layer was double to number of inputs to reach relatively small network size. The

maximal iteration number of the training process was limited to 2000 in every tested network configuration. The final model from actual training process was selected using minimal mean squared error (MSE) (Füvesi, 2011) value of validation dataset. Widrow – Nguyen algorithm was used to initialize the value of the weights in the networks (Nguyen, 1990), which speeded up convergence to a minimum of in the training process of the ANN. Rprop training algorithm (Riedmiller, 1993) was used for the training of the network. The training and validating dataset were generated from measurements but the validation set was differing from training set. The datasets includes the result of measurement in which dynamically changed load was applied during training process. The main features of dataset can be found in the Table 2.

Table 2. Range of different measured parameters

Parameters	Training dataset	Validating dataset
Size of dataset [pcs]	55050	7300
Number of measurements, which was used in generation [pcs]	9	1

From the numerous models different selection criteria (SC) were applied like: Akaike information criterion (AIC) (Akaike, 1974), Bayesian information criterion (BIC), final prediction error (FPE) (Pizarro, 2000), MSE, and Pearson’s correlation coefficient (PCC) (Brown, 1994) to select the best model. The AIC, BIC and FPE are based on MSE value and apply any type of penalty function. The penalties are basically the function of different parameters of the actual neural network like number of weights in network, size of dataset, etc. PCC characterizes the strength of regression between the required and estimated output (Brown, 1994).

To make our later work easier same structured velocity and torque observer was selected. An evaluation function (EF) is introduced to select the best model pair, which can be calculated from the squared root of the normalized form of previously listed criteria (Eq. 3).

$$EF_{sc} = \sqrt{\text{norm}^2(SC_{\text{velocity}}) + \text{norm}^2(SC_{\text{torque}})} \tag{3}$$

Normalized EFPC can be seen on Figure 2. The best model emerges from NNARX structure. The model has five time delay per input type and delay is 110.

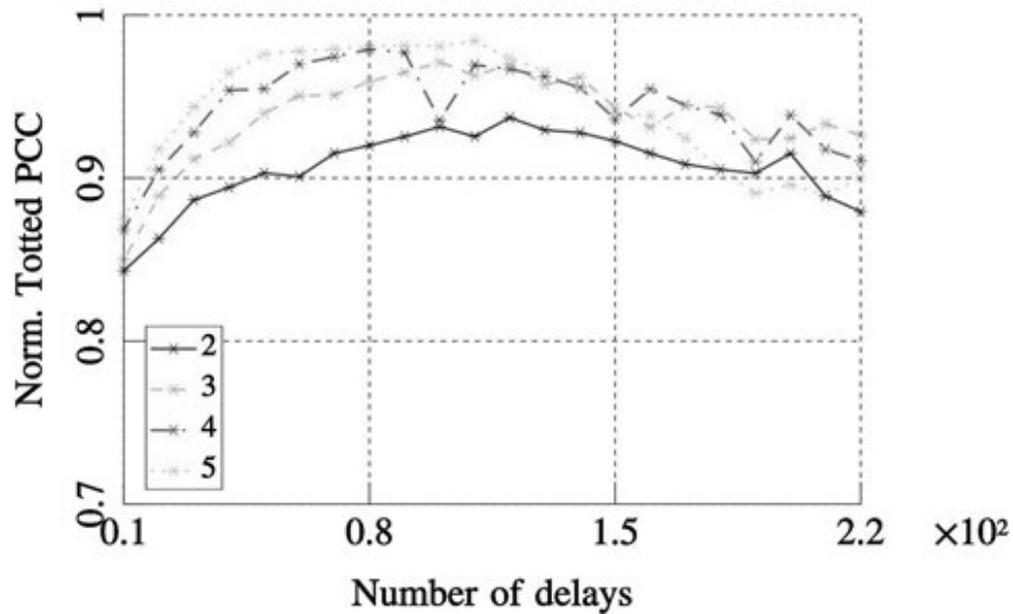


Fig. : Normalized totted PCC SC for the models

#### 4. DEVELOPING THE FAULT DIAGNOSTIC NETWORK

The development of fault separator neural networks structure includes more steps such as: faulty signal generation for the neural networks, analysing the different neural network structure for the diagnoses; training the different structured networks, evaluation of the model, find the best model.

##### 4.1 Structure of the fault separator

Using the developed models fault separation and identification were done. These two steps can be called fault diagnose (Isermann, 2011). A feedforward neural network based separator structure were developed which was able to make the fault diagnoses. The structure has four outputs (F1, F2, F3, F4), which describe appearance of fault in the system. Every output describes a fault type in the system.

- F1: voltage signal fault;
- F2: current signal fault;
- F3: fault in velocity signal;
- F4: fault in torque signal.

If the output signal is not equal with zero than the system includes some fault. The occurred fault size is proportional with the size of the signal of the network. The sign of the signal has also some meaning. If the signal is less than zero than the occurred faulty signal is less than in normal conditions. In the case of positive deviation, the level of faulty signal is higher correlated to signal in normal conditions.

Four different input configurations were analysed NNFIR, NNARX, in two other cases the NNFIR and NNARX structures were supplemented with the difference of normalized form of the measured signals from output side of actuator (torque, velocity) and the output of the suitable observers.

So the input space includes more signals: the normalized voltage and current signals, the normalized velocity and torque signals were used beside the earlier developed outputs of observers. In adequate cases the observers were fed with the same voltage and current signals. The results of the diagnoses required to

raise the TDL of a part of the inputs. The different TDL configurations were investigated to find most suited structure to the task from analysed model space.

## **4.2 Fault generation**

The faulty signals were generated artificially using the signal from the earlier described measurements. The generated faults appear in the signals like suddenly occurred offset on the signal. During the generation of signals the separated faults were changed from -30% to 30% percentage of the maximal value of actual signal type in normal condition.

Two longer independent dataset were generated, which were used for training and validation purpose. A third dataset series was created for evaluation case to test the performance of the developed networks. This third dataset includes more fault cases, where once only one type of fault is occurred. These sets include three, different sized faults which are sparsely in time occurred.

## **4.3 Training the NNs**

Using the dataset more feedforwarded neural networks were trained with different structures. The training process was a special active training method, where the training data was generated during the training process. The advantages of the method are in the shorter computation time of network and the less memory usage compared to the normal method. The structure of active training algorithm can be seen on Figure 3.

First the parameters of the NN must be set, like number of input, number of hidden neurons, e.g. After that comes the generation of a base training and validation set. During the training process the size of training epoch is dynamically changed. The period of dataset change was also analyzed and found that good efficiency can be achieved if the set is recreated in every 50 training iteration.

Using the actual dataset a training iteration is come, which use the actual training dataset to train the network. At the end of the process comes the calculation of MSE of validation epoch, which value can be used to save best network, during the training process and avoid overlearning. Finally if the iteration number reached its maximum value the process ended otherwise the next iteration is come.

Using the method more than 20 percent time saving can be achieved networks with smaller input size. The method was tested using mainly classification problem and the connected fault diagnostic problem. The Figure 4 shows the algorithm efficiency in the function of the number of the inputs of the diagnostic network.

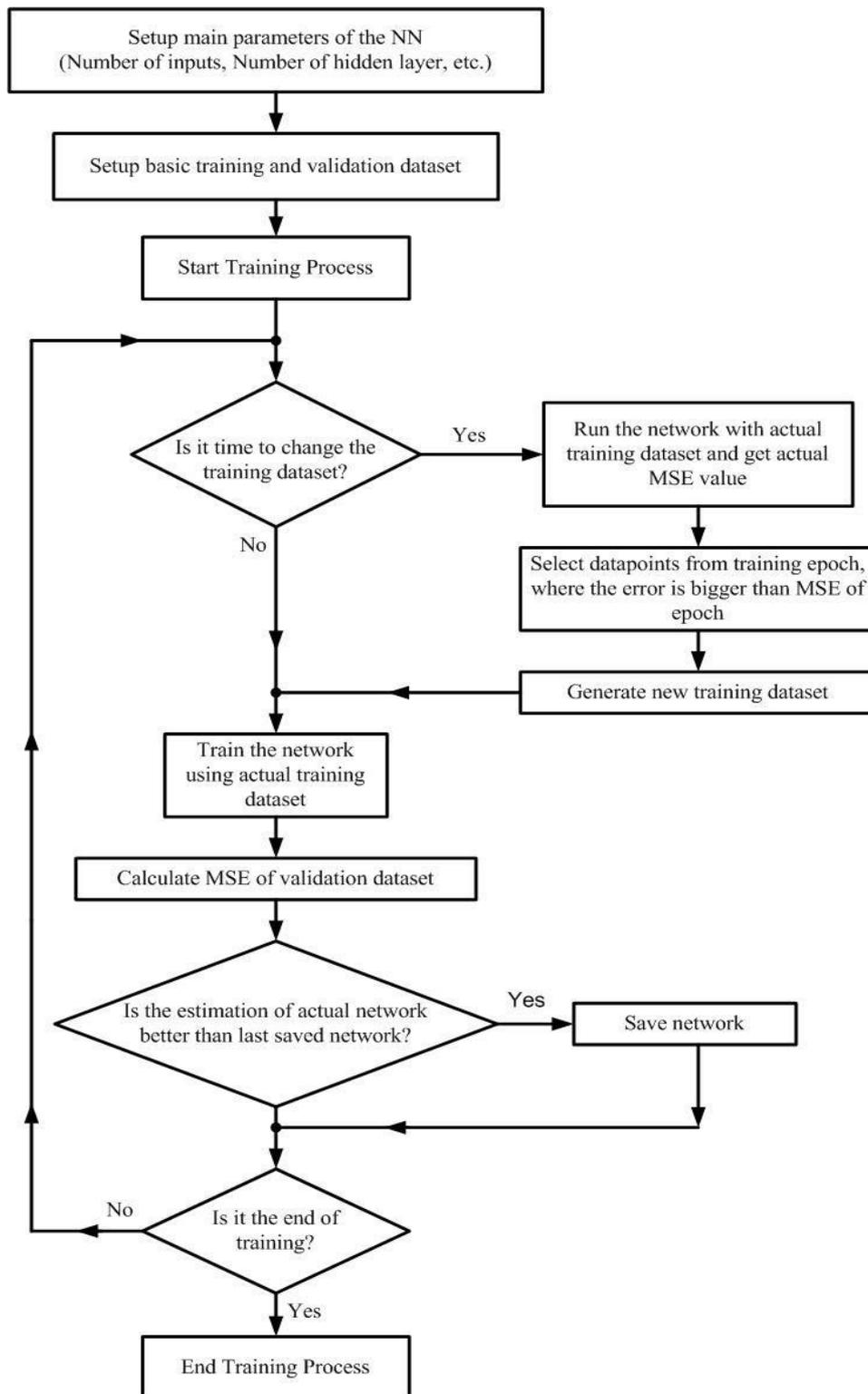


Fig. : Active learning algorithm

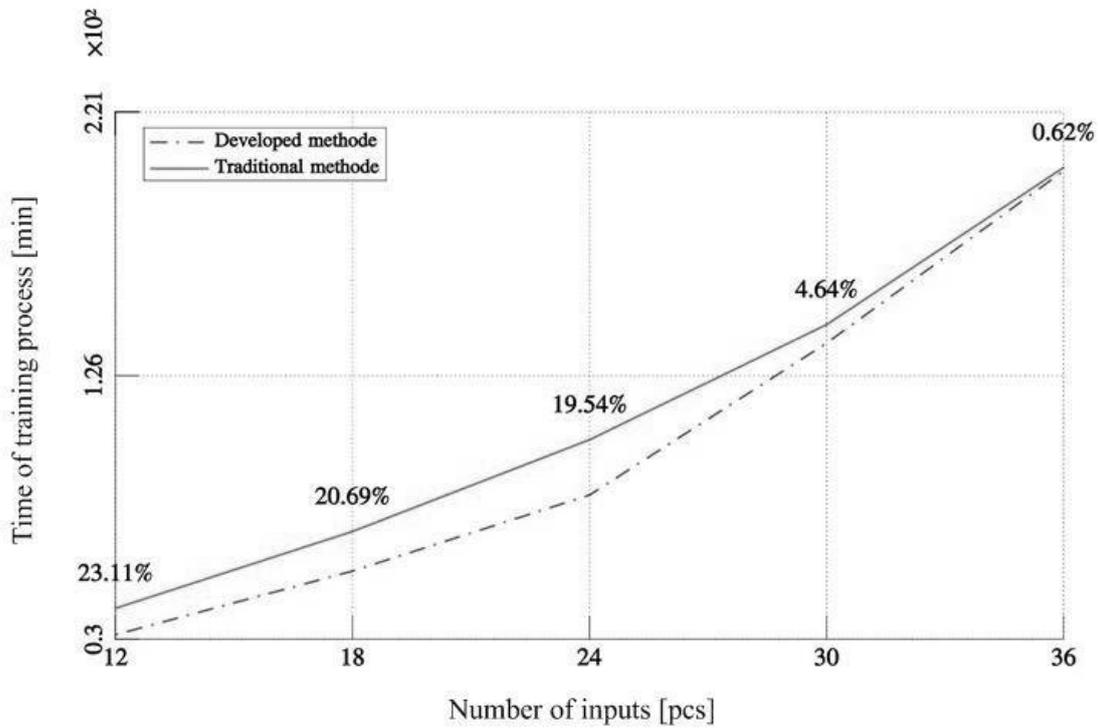


Fig. : Compare the training times of active and traditional training algorithms in the function of inputs of separator network e learning algorithm

## 5. EVALUATION OF THE EFFICIENCY OF THE STRUCTURE

Several networks were trained and a complex evaluation process was developed to analyse the performance of different structured networks. The previously describe test datasets were used to test the performance of the networks.

The next main parameters were investigated: a) the ability to give signal on correct channel on proper time; b) how many correct the found fault size compared to the generated signal.

### 5.1 Edge recognition ability

In a diagnostic system, perhaps the most important attribute to provide the minimum number of pseudo alarm. It must generate alarm signals when and where the faults occur.

To investigate the performance of a fault diagnostic system the MSE value does not give adequate information about the efficiency of systems, thus some other index value usage is suggested. The faults to detect have the property of shorter or longer impulse shaped. These impulses can be characterize with the occurrence and disappearance on the signal. If we compare the rising and falling edges of the real signal with the result of diagnostic systems, the discovered edges can be sorted in three categories. The edges which were on time detected are the recognized edges (RE). Other part of the recognized edges is the missed one, which means that is a change on the signal but it was not detected by the diagnostic system. These edges are the missed edges (ME). Last category is the false alarms. These edges were detected by the diagnostic system but in fact there is no fault in the system (BE).

Using the parameters an index value can be proposed called edge recognition ability (Eq. 4). This quantity can be used to rate the performance of the separator network.

$$\text{ERA} = \text{RE} / (\text{RE} + \text{ME} + \text{BE}), [\%]$$

(4)

To calculate the ERA edges must be found in the output signal of diagnostic neural network structure. There are several methods, which can be used to find edges in one dimension. One traditional method is, where after some filtering the signal derivatives are analysed to find the step like changes in the signal (Füvesi, 2012). Drawback of the method is the sensitivity to the noises. We applied the method but it gave poor efficiency in our case. The other used method is the Canny edge detection method, which use the first derivative of Gaussian to approximate the optimal finite length filter (Canny, 1986). This method gives much better result. The Figure 5 shows the different diagnostic structures efficiency using the totted ERA value for the different cases using the Canny edge detection method.

The best result was achieved using the NNARX separator structure. The additional difference inputs, in the case of “FIR and error” and “ARX and error”, do not improve the result as it was previously expected. Using the FIR structure gave poor result in both cases. The best result was about 80% of the edges was detected it means more the 1100 good detection from 1440 cases. The Figure also shows the different TDL sizes which rises the size of the network and do not increase the performance of the network.

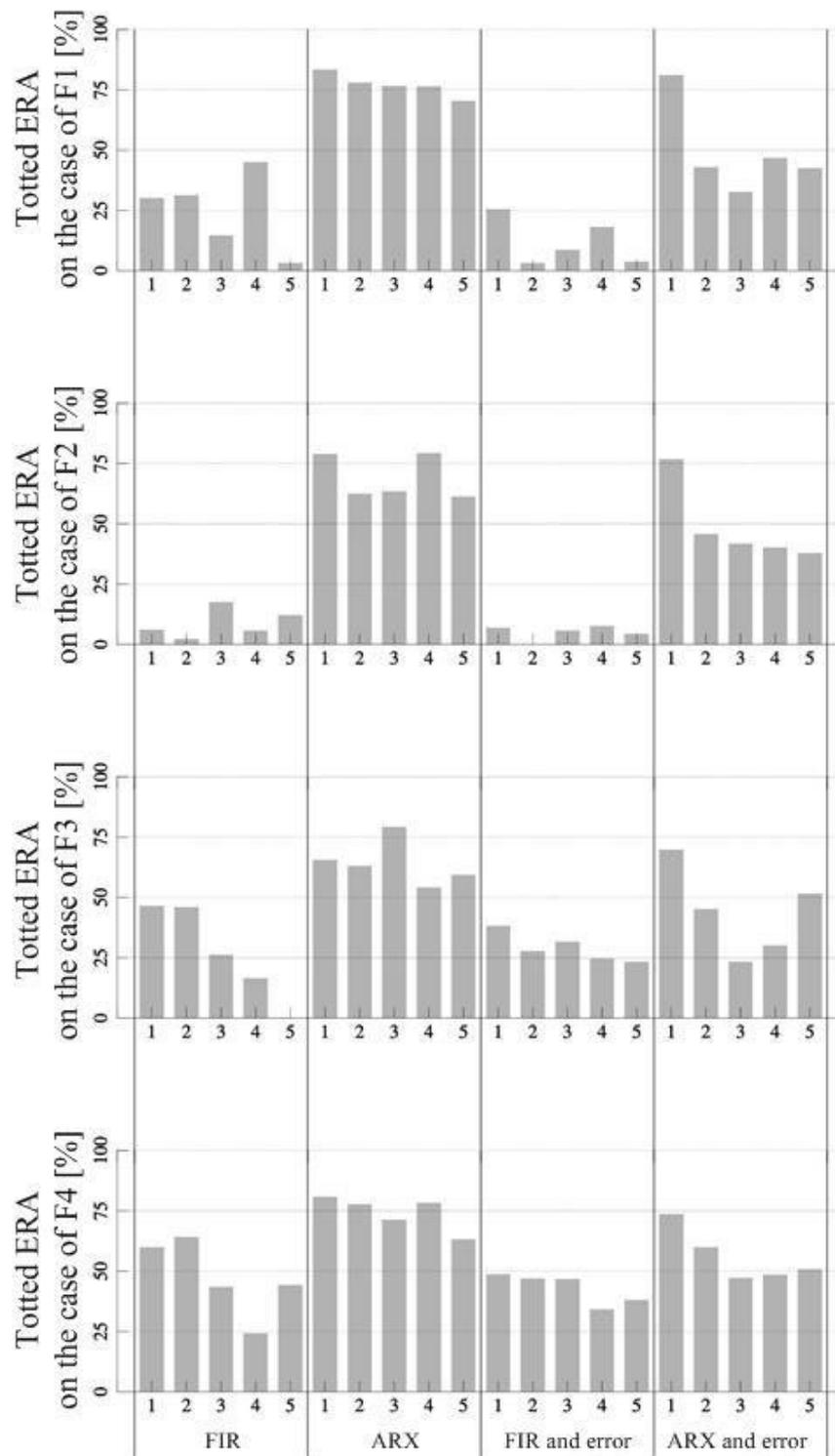


Fig. : Totted ERA of different separator structures

## 5.2 Size of found edges

The separator structure was able to estimate the size of fault occurred in the system. To analyse this, relative error of the estimation from the actual real fault signal is calculated. The separator structure generate about 15-20% average error in the investigated error size. The relative error and standard deviation

can be large when small errors (less 5% is the error of full range of signal) are estimated. Figure 6. shows the results of the structure, which was the best in previous ERA investigation.

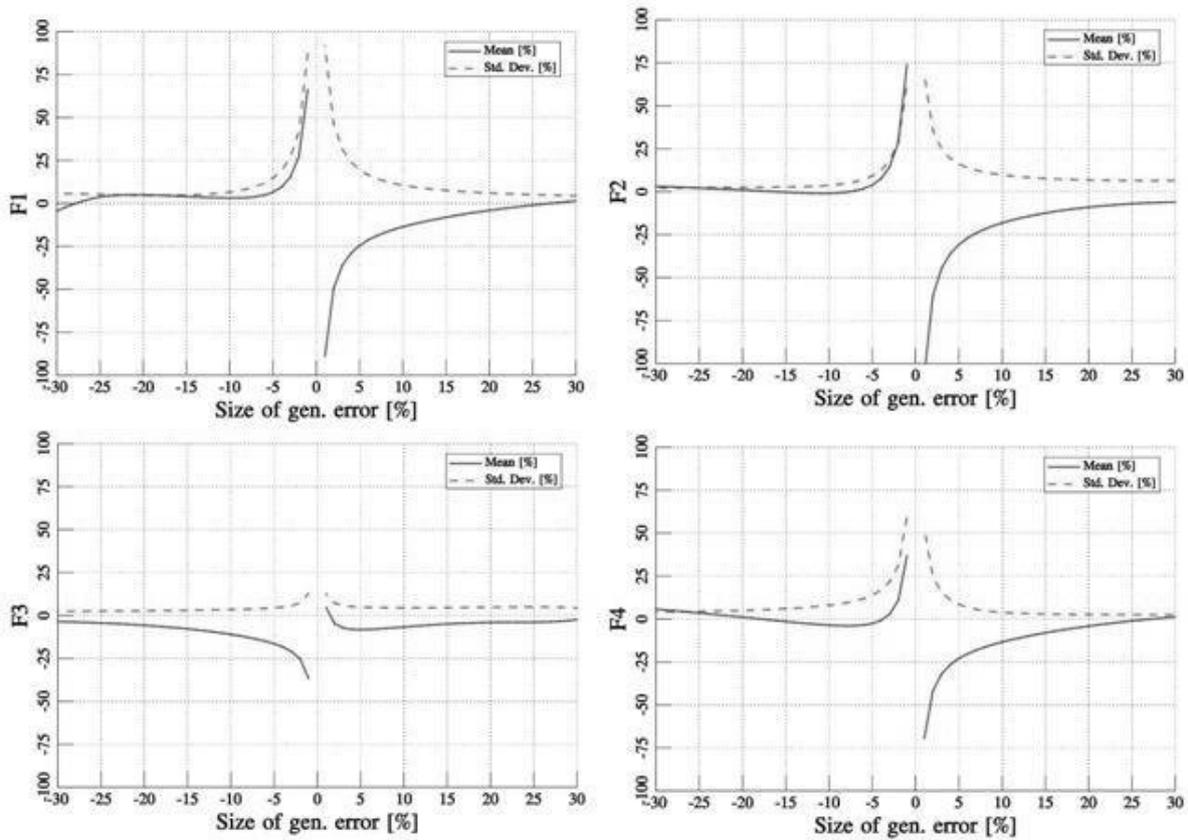


Fig. : Mean and standard deviation of error of estimation of the best separator structure in the function of the artificially generated fault size in the cases of different faults

## 6. SOFTWARE AND HARDWARE BACKGROUND

During the research task oriented, time optimised C/C++ programs were developed for data pre and post processing, neural network training, evaluate the results. The reason of the development was that the script based, open source mathematical software packages (eg. Scilab, Octave) can be quit slow during the training process of ANN or at complicated file operations in which more files must be open and process.

For program compiling GCC was used with some useful frameworks and libraries. FANN neural network library was used for the neural network programming (Nissen, 2003), MathGL (Balakin, 2008) graphical library for the graphs, GSL (Galassi,2009) was applied for the function library and Armadillo (Sanderson, 2010) for matrix and vector calculation and operation.

As hardware a Core2Duo E8400@3GHz based PC was used with 4GB RAM on Fedora 18 OS.

## 7. CONCLUSIONS

This paper shows an example to that the artificial algorithms and methods are well usable to solve complex tasks, such as fault separation and diagnoses in one structure. The advance training method reduces the computation time. The short TDL sized NNARX MLP models and separator structures give appreciably good result. The detailed evaluation procedure shows that, the used mathematical structure operates well in a width operation range.

## ACKNOWLEDGEMENT

This research was partially carried out in the framework of the Center of Excellence of Mechatronics and Logistics at the University of Miskolc.

## REFERENCES

- Zhou Y.; Hahn J.; Mannan M. S. (2003). Fault Detection and Classification in Chemical Processes based on Neural Networks with Feature Extraction, *ISA Transactions* 42, pp.651-664.
- Azhar S.; Rahman S. A. B. (2010). Application of Artificial Neural Network in Fault Detection Study of Batch Esterification Process, *Int. Journal of Engineering and Technology IJET-IJENS Vol: 10 N: 03*, pp.36-39.
- Nicholson A. E.; Brady J. M. (1992). Sensor Validation using Dynamic Belief Networks, *Proc. of the 8th Conference on Uncertainty in Artificial Intelligence*, p. 207.
- Spina P. R. (2000). Reliability in the Determination of Gas Turbine Operating State, *Proc. of the 39th IEEE Conf. on Decision and Control*, Sydney, NSW, Australia, p. 2639.
- Pranatyasto T. N.; Qin S. J. (2001). Sensor Validation and Process Fault Diagnosis for FCC Units under MPC Feedback, *Contr. Eng. Practice* 9, p. 877.
- Zhang Y.; Dudzic M.; Vaculik V. (2003). Integrated Monitoring Solution to Start-up and Run-time Operation for Continuous Casting, *Annu. Rev. Contr.* 27, p. 141.
- Mehranbod N.; Soroush M.; Panjapornpon C. (2005). A Method of Sensor Fault Detection and Identification, *Journal of Process Control* 15, pp. 321–339.
- Jabbari A.; Jedermann R.; Lang W. (2007). Application of Computational Intelligence for Sensor Fault Detection and Isolation, *World Academy of Science, Engineering and Technology* 33, pp. 265-270.
- Zhang J. (2006). Improved on-line Process Fault Diagnosis through Information Fusion in Multiple Neural Networks, *Computers & Chemical Engineering* 30, pp. 558-571.
- Amann P.; Perronne J. M.; Gissingner G. L.; Frank P. M. (2001). Identification of Fuzzy Relational Models for Fault Detection, *Control Engineering Practice* 9, pp. 555-562.
- Padmakumar S.; Agarwal V. (2009). A Comparative Study into Observer based Fault Detection and Diagnosis in DC Motors: Part-I, *World Academy of Science, Engineering and Technology* 51, pp.318-323.
- Sainz Palmero G. I.; Juez Santamaria J.; Moya de la Torre E. J.; Peran Gonzalez J. R.(2005). Fault Detection and Fuzzy Rule Extraction in AC Motors by a Neuro-Fuzzy ART-based System, *Engineering Applications of Artificial Intelligence* 18 pp.867–874.
- Kolla S.; Varatharasa L. (2000). Identifying Three-phase Induction Motor Faults using Artificial Neural Networks, *ISA Trans.* 39, pp. 433–439.
- Ciurys M.; Dudzikowski I.; Gierak D. (2009). Modelling of a Car Starter with Permanent Magnet Commutator Motor, *COMPEL The Int. Journal for Computation and Mathematics in Electrical and Electronic Engineering*, Vol. 28 Iss: 3, pp.722-729.
- Füvesi V.; Kovács E. (2011). Modeling Loaded Starter Motor with Neural Network, *Proc. of 12th IEEE Int. Symposium on Computational Intelligence and Informatics, (CINTI2011)*, Budapest, Hungary, ISBN 978-1-4577-0044-6, pp. 551-554.
- Norgaard M.; Ravn O.; Poulsen K. N.; Hansen L. K. (2000). *Neural Networks for Modelling and Control of Dynamic Systems*, Springer-Verlag, London, pp 18-34.

- Nguyen D.; Widrow B. (1990). Improving the Learning Speed of 2-layer Neural Networks by Choosing Initial Values of the Adaptive Weights. IJCNN, vol III, pp 21-26.
- Riedmiller M.; Braun H. (1993). A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP algorithm, In Proc. of the IEEE Int. Conf. on Neural Networks, San Francisco, CA, pp. 586-591.
- Akaike H. (1974). A New Look at the Statistical Model Identification, IEEE Trans. on Automatic Control, vol. 19, pp 716-723.
- Pizarro J.; Guerrero E.; Galando P. L. (2000). A Statistical Model Selection Strategy Applied to Neural Networks, European Symposium on Artificial Neural Networks, pp. 55-60.
- Brown M.; Harris C. (1994). Neurofuzzy Adaptive Modelling and Control, Prentice Hall, London.
- Isermann R. (2011). Fault diagnosis applications: model based condition monitoring, actuators, drives, machinery, plants, sensors, and fault-tolerant systems, ISBN 978-3-642-12766-3, Springer.
- Füvesi V.; Kovács E. (2012). Separation of Faults of Electromechanical Drive Chain using Artificial Intelligence Methods, 18th "Building Services, Mechanical and Building Industry days" Int. Conf., Debrecen, Hungary, pp. 19-27.
- Canny J. (1986). A Computational Approach to Edge Detection, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 8, No. 6, November, pp. 679-698.
- Nissen S. (2003). Implementation of a Fast Artificial Neural Network Library (fann), Department of Computer Science University of Copenhagen (DIKU).
- Balakin A. A. (2008). MathGL Manual (for version 1.11.2), <http://mathgl.sourceforge.net>, GNU Free Documentation.
- Galassi M. (2009). GNU Scientific Library Reference Manual - Third Edition, ISBN 0954612078, GNU Manual, Network Theory Ltd.
- Sanderson C. (2010). Armadillo: An Open Source C++ Linear Algebra Library for Fast Prototyping and Computationally Intensive Experiments, Technical Report, NICTA.