Modelling and Prediction of Surface Roughness in CNC Turning Process using Neural Networks

Tomislav ŠARIĆ*, Đorđe VUKELIĆ, Katica ŠIMUNOVIĆ, Ilija SVALINA, Branko TADIĆ, Miljana PRICA, Goran ŠIMUNOVIĆ

Abstract: The paper presents an approach to solving the problem of modelling and prediction of surface roughness in CNC turning process. In order to solve this problem an experiment was designed. Samples for experimental part of investigation were of dimensions $\phi 30 \times 350$ mm, and the sample material was GJS 500 - 7. Six cutting inserts were used for the designed experiment as well as variations of cutting speed, feed and depth of cut on CNC lathe DMG Moriseiki-CTX 310 Ecoline. After the conducted experiment, surface roughness of each sample was measured and a data set of 750 instances was formed. For data analysis, the Back-Propagation Neural Network (BPNN) algorithm was used. In modelling different BPNN architectures with characteristic features the results of *RMS* (Root Mean Square) error were controlled. Specially analysed were the *RMS* errors realised by different number of neurons in hidden layers. For the BPNN architecture with one hidden layer the architecture (4 – 8 - 1) was adopted with *RMS* error of 3,37%. In modelling the BPNN architecture with two hidden layers, a considerable amount of architectures was investigated. The adopted architecture with two hidden layers (4 - 2 - 10 - 1) generated the *RMS* error of 2,26%. The investigation was also directed at the size of the data set and controlling the level of *RMS* error.

Keywords: CNC turning; Neural Networks; prediction; surface roughness

1 INTRODUCTION

At the present time of the fourth industrial revolution contemporary manufacturing processes integrate methods of artificial intelligence. The methods of artificial intelligence are the tools that enable intelligent production i.e. they make it possible for technologists to plan technological processes quicker and more efficiently. As every technological process is planned among others also according to the requirements of quality, the quality of machined surface (surface roughness) needs to be also pointed out. The quality of machined surface is directly correlated with the manufacturing process which usually contains various influencing parameters (variables). The manufacturing processes being multivariable, they are usually hard to be modelled as optimal. While modelling and investigating the surface roughness dependence on the parameters of a manufacturing process, the investigators designed various models. In paper [1] authors investigated the surface roughness prediction for turning operations using computer vision and artificial neural networks (ANN) with evolutionary algorithm. For the purposes of investigation, they designed a model in which the surface roughness was the output parameter while the model input parameters were: cutting speed, feed, depth of cut and average grey level of the surface image of the workpiece (acquired by computer vision). Computer vision and soft computing approach were also used in paper [2] as the methodology for recognizing the errors of surface roughness in the CNC turning process. For the purposes of investigation a model was designed which contained the following input parameters: feed, depth of cut, cutting speed, frequency range, grey scale value. Based on the given model, training and implementation of neural networks, an efficient methodology was obtained for discovering the error rate of surface roughness in the CNC turning process. A predictive model for various kinds of materials (austenitic, martensitic and duplex stainless steels) in CNC turning process was designed in paper [3] for investigation with the following input parameters: cutting speeds (120, 150, 180 and 210 m/min), feed (0,1 mm/rev) and depth of cut (1 mm) and using coated

cemented carbide tools. The following parameters were defined as the model output parameters: cutting forces and surface roughness. Based on the proposed model and the parameters obtained during experimental work these were used in the process of the application of ANNs and for the comparison of obtained results.

In modelling and optimization of the machining parameters in the turning process, the authors of paper [4] proposed a model with the following input parameters: tangential cutting force, cutting power and the material removal rate, and with surface roughness as the output parameter. In experimental part of collecting data during the turning process, coated and uncoated silicon nitride ceramic tools were used while for the process of prediction the approach using neural networks and response surface methodology (RSM) was applied. In the process of optimization genetic algorithm (GA) was used. It was proven that a coated ceramic tool provides better surface quality and minimal cutting force in comparison with those obtained with an uncoated ceramic tool. Authors [5] use ANNs in estimating surface roughness by forming a model in which the machining parameters, cutting forces, sound and vibrations of the turning process were used as input parameters. Regression models were used for comparison. It was proven that neural networks estimated the state of surface roughness with more than 98% accuracy in relation to the formed regression models with more than 90% accuracy.

Comparison of three machine learning methods was suggested by authors [6] for prediction of output parameters of high speed turning process. The analysed output parameters were surface roughness (Ra), cutting force (F_c), and tool life (T). Different methods of machine learning were used for the process modelling: Support Vector Regression (SVR), polynomial (quadratic) regression, and ANNs. The best results in predicting F_c and Ra were obtained by polynomial regression while in predicting T the best results were obtained by ANN. The investigation presented in [7] dealt with the obtaining of surface roughness by the use of fuzzy inference system (FIS) and comparing the results, among others, with the results derived by ANNs. Modelling and prediction of

surface roughness and cutting forces in the process of finish turning with mixed ceramic tool using ANNs was suggested by authors [8]. For this research, a model with the following input parameters was used: cutting speed, depth of cut, feed and tool nose radius. During experiments authors used different materials. In paper [9] machining of aluminium alloy is suggested as well as modelling and prediction of the turning process using ANNs. For experimental part the following parameters were selected for the model: surface roughness, cutting forces, cutting temperature, material removal rate, cutting power, and specific cutting pressure. After the performed experimental work and collected data a neural network was modelled that estimated the process cutting performance with high precision.

By using the models, different parameters were investigated that influenced output parameters of the model. Among them surface roughness was certainly an important parameter. The various demonstrated models confirmed the problem complexity. Some authors [10, 11] use these models to demonstrate the problem of tool wear. In addition to the quoted papers in the field of turning, ANNs are used in combination with other methods (classification and regression tree, support vector machine models, GA, adaptive particle swarm optimization algorithm) for modelling and prediction of surface roughness [12, 13]. A special accent is given to the neural networks Back-Propagation (BPNN) algorithm for prediction of surface roughness [14], estimation of machining time [15], optimization of cutting parameters in combination with genetic algorithm [16] and prediction of cutting forces [17]. Investigation of various algorithms of neural networks (Back-Propagation Neural Network, Modular Neural Network and Radial Basis Function Neural Network) connected with the problem of surface roughness prediction was conducted by authors in paper [18]. Paper [19] describes the development of a model for surface quality prediction based on Radial Basis Function Neural Network (RBFNN). The RBFNN model results were compared with the BPNN model in view of the computing speed and accuracy. Authors in [20] use RBFNN for surface roughness prediction in hard turning process. Modelled RBFNNs showed the capability to foresee surface roughness in an exact, precise and accessible way. In paper [21] authors proposed to apply hybrid evolutionary neural-fusion system for evaluation of surface roughness. The proposed system also included, among others, prediction of surface roughness by cutting parameters and control of obtained or needed surface roughness by means of the characteristics quantified from the digital image of the observed machined surface. Authors [22, 23] use various algorithms of neural networks for investigation of manufacturing problems.

In all of these papers conclusions in most cases provide justification for the application of neural networks in investigating and designing models for prediction of surface roughness.

The present paper deals with modelling and predicting surface roughness in CNC turning process using the designed model and experimental results of different architectures and properties of neural networks based on the Back-Propagation algorithm.

2 METHODOLOGY AND AIM OF INVESTIGATION

The aim of every manufacturing process is to achieve the required (demanded) quality of a product with acceptable characteristics of the process, such as: production costs, terms of delivery, machines and available tools. The quality of a product is defined, among others, also by the surface roughness which is required for the product to be functional. During chip-forming machining surface roughness is in the function of machining parameters. Technologists in production firms use various approaches to obtain the required surface roughness choosing optimal parameters and satisfying all the earlier mentioned characteristics. Different approaches to modelling process characteristics in the function of surface roughness are given in the literature review.

For the research presented in this paper a model is proposed with the following input parameters (variables): vector X_i = {cutting speed v_c (m/min), feed f (mm/rev), depth of cut a_p (mm) and different cutting inserts}, and the following output parameter (variable): vector Y_i = {mean arithmetic deviation of profile Ra}.

If the middle line system is taken as the referent system that is used to evaluate the profile and if the middle line m is taken as the referent line, the middle line m divides the profile so that within the referent line l the sum of squares of all the profile y deviations from this line is the smallest. According to the defined system (Fig. 1) two magnitudes are defined: Ra - Mean arithmetic deviation of the profile and Rq - Mean square deviation of the profile.
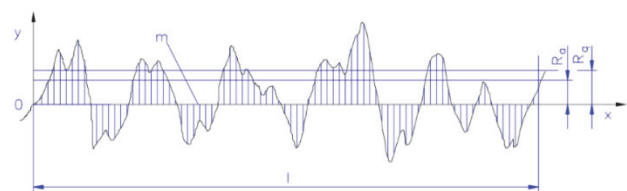

Figure 1 Definition of magnitudes Ra and Rq

Ra - The mean arithmetic deviation of the profile is the mean arithmetic value of absolute profiles y within the borders of the referent line l, given by the Eq. (1).

$$Ra = \frac{1}{l}\int_0^l \left| y(x) \right| \mathrm{d}x \tag{1}$$

Rq - The mean square deviation of the profile is the profile mean square value within the borders of the referent line l, and is given by the Eq. (2).

$$Rq = \sqrt{\frac{1}{l}\int_0^l y^2(x)\,\mathrm{d}x} \tag{2}$$

2.1 Design of Experiment and the Experimental Work

Designed experiment for the present research was performed with six cutting inserts. Three cutting inserts were standard ones and three were of Wiper geometry (their geometry was identical to the standard ones-

condition for comparison). During the experiment, input variables were acquiring different levels:

- Cutting speed - v_c: 200; 225; 250; 275 and 300 m/min
- Feed - f: 0,1; 0,2; 0,3; 0,4 and 0,5 mm/rev
- Depth of cut - a_p: 0,5; 1; 1,5; 2 and 2,5 mm.

Samples for analysis of the suggested model were machined on CNC lathe (DMG Moriseiki- CTX 310 Ecoline) in the process of longitudinal external turning. Dimensions of samples were $\phi 30 \times 350$ mm, and the sample material was GJS 500 - 7.

Based on thus defined design of experiment 125 experiments ($5 \times 5 \times 5$) were conducted with the corresponding measurement *Ra* for every cutting insert. As six cutting inserts were defined in the design of experiment, the total base was composed of 750 instances (each instance is a combination of the levels of the input variables with the corresponding measurement *Ra*). Tab. 1 displays a part of the data set.

Table 1 Part of the data set

v_c / m/min	f / mm/rev	a_p / mm	Insert	*Ra*
200	0,1	0,5	1	1,26
225	0,3	1,5	3	5,87
250	0,5	1,5	1	31,42
225	0,5	0,5	2	10,49
200	0,1	2	4	0,53
225	0,3	2	4	1,99
275	0,3	0,5	5	3,79
300	0,1	0,5	6	0,41
250	0,1	1,5	5	0,59
200	0,3	0,5	1	11,27
225	0,1	1,5	1	1,30
200	0,3	1	3	5,78
275	0,3	2	5	4,61
300	0,3	2,5	6	1,64
275	0,2	1	2	2,65
200	0,4	0,5	1	20,11
300	0,3	1	2	3,96
300	0,2	0,5	4	1,35
275	0,4	1	5	6,95
300	0,5	2	6	3,98

2.2 Preparation of Data for Experimental Work with Neural Networks

Data set of 750 instances for experimental work with different algorithms of neural networks is based on experimental data obtained by the designed experiment. The data set was randomized and divided in three subsets according to the output variable by applying the principle: 60% for training, 20% for testing and 20% instances for validation (Fig. 2).
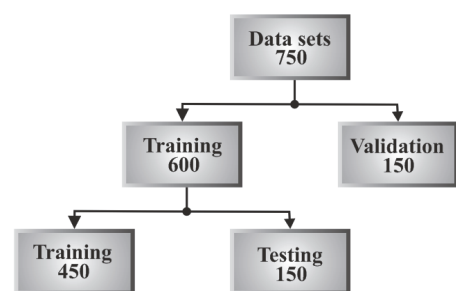
Figure 2 Division of data set for experimental work with NN

The modelled data subsets were used in experimental work for each of the modelled architectures of a neural network. Modelling of the neural networks' various architectures is manifested in investigation of optimal attributes. For every neural network's modelled architecture with a defined algorithm a training process with testing is carried out. The process of investigation and the experimental work are displayed in an iterative procedure and in the recognition of the dependence of individual attributes on a particular output of the set model. As a criterion for a successful architecture design of neural networks the criterion of *RMS* error was chosen (*Root Mean Square Error-RMSE* or *RMS*), given by the Eq. (3).

$$RMS = \sqrt{MS} = \sqrt{\frac{\sum_{n=1}^{N}(d_n - y_n)^2}{N}}$$

(3)

where: *MS* - Mean Square error; *N* - Number of pairs of the training set input-output values; y_n - Neural network n-th output; d_n - Desired value of a neural network n-th output.

2.3 Back-Propagation Neural Network-General Approach

The Back-Propagation Neural Network (BPNN) is one of the most often used neural networks in investigations. In the basic architecture a neural network always has an input layer, an output layer and at least one hidden layer. There is no theoretical limit to the number of hidden layers, but the usual number is one or two. The analysed literature makes it clear that maximally four layers (three hidden and one output layer) are necessary for solving the arbitrarily complex problems of the classification of samples. In architecture, every layer is usually completely connected with the next layer. Fig. 3 displays the usual common structure of a neural network with one hidden layer. The arrows mark the information flow during passage through the network. By passing through the network, the computed output values are compared with the real ones and an error is being computed. The computed error is propagated back through the network. By this procedure the weights of the connections between the neurons are being changed, and the process of changing the connections' weights makes the training process of a neural network possible.
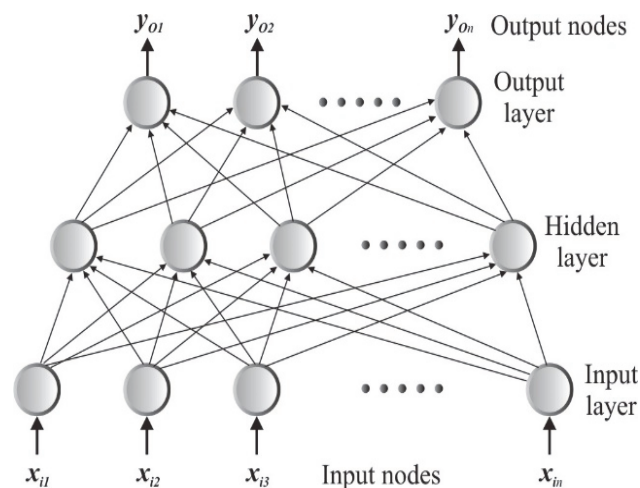
Figure 3 Typical Back-Propagation Neural Network

Basic element of a neural network is the neuron. A biological neuron is modelled and as it can be seen from Fig. 4 the neuron body takes the role of a summation function while the inputs into the summation function take the role of dendrites. The level of activation sensitivity of the biological neuron is taken over by transfer function which defines the moment of sending (firing) the impulse to the neuron output. The transfer function can be linear or non-linear. With linear transfer functions the output from the summation function is multiplied with some factor and the obtained sum is transferred to the neuron output. With non-linear functions the summation function outputs are changed in accordance with various forms of functions, and the neuron outputs can assume different values depending on transfer function. The information flow through neural network being usually completely connected, clear marking is necessary due to the training rules' description. In the exponent a layer of network is usually marked in which the training process is computed. The marking is given as follows:

- $x_j^{[s]}$ - current output state of j-th neuron in layer s

- $w_{ji}^{[s]}$ - weight on connection joining i-th neuron in layer $(s-1)$ to j-th neuron in layer s

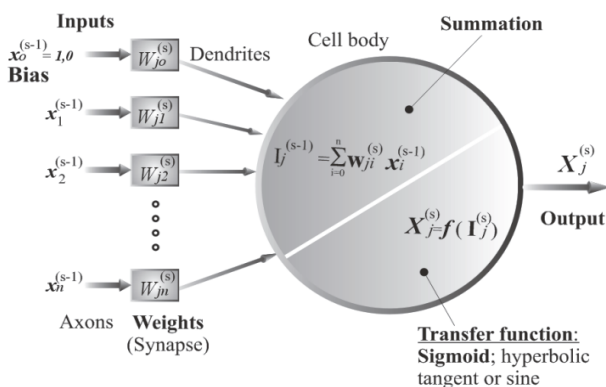- $I_j^{[s]}$ - weighted summation of inputs to j-th neuron in layer s.

Figure 4 Typical Back-Propagation processing element (neuron)

A Back-Propagation element therefore transfers its inputs as follows:

$$x_j^{[s]} = f\left(\sum_i \left(w_{ij}^{[s]} \cdot x_i^{[s-1]}\right)\right) = f\left(I_i^{[s]}\right) \qquad (4)$$

where f is usually a Sigmoid transfer function but can be any differentiable function. The Sigmoid transfer function is defined by Eq. (5). In the present research the Sigmoid transfer function for the BPNN modelling is adopted.

$$f(z) = \left(1.0 + e^{-z}\right)^{-1} \qquad (5)$$

3 EXPERIMENTAL WORK AND ACHIEVED RESULTS

Experimental work on modelling and researching acceptable architecture of a neural network was carried out on adopted data set and selected Sigmoid transfer function. The reason for adopting the Sigmoid function was the data

modelling and the results achieved in previous investigations. The following learning rules (algorithms) were adopted for modelling the neural networks architectures: Delta, Delta-Bar-Delta (DBD), Ext. Delta-Bar-Delta (Ext. DBD) and Normalized Cumulative Delta (Norm. Cum. Delta). Then the modelling and defining the properties of architecture of a neural network with one hidden layer was carried out. Tab. 2 and Fig. 5 display the achieved results with the level of *RMS* error in relation to the number of neurons in the hidden layer. It can be seen from Tab. 2 that the lowest level is the one of *RMS* error with eight neurons in the hidden layer and it is 3,37%. The other rules (algorithms) of training were also investigated based on the adopted architecture. The achieved results are as follows:

- Delta training rule *RMS* = 4,82%
- Delta-Bar-Delta training rule *RMS* = 4,73%
- Norm. Cum. Delta training rule *RMS* = 4,80%.

Table 2 *RMS* (%) in function of number of neurons

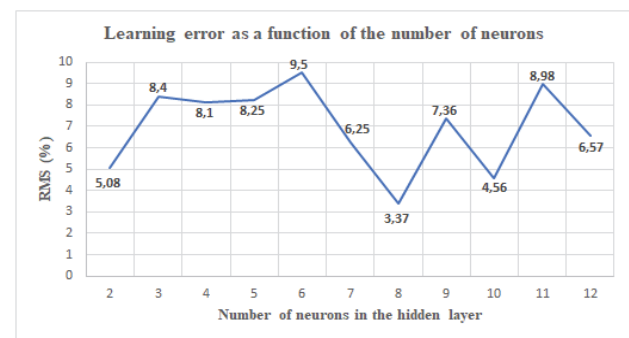| Number of neurons	2	3	4	5	6	7
RMS / %	5,08	8,40	8,10	8,25	9,50	6,25
Number of neurons	8	9	10	11	12	
RMS / %	3,37	7,36	4,56	8,98	6,57	

Figure 5 *RMS* errors in the function of the number of hidden neurons

From the realised lowest *RMS* error the architecture with eight hidden neurons is adopted i.e. the (4 – 8 - 1) architecture (Fig. 6) along with the combination of Sigmoid transfer function and extended Delta-Bar-Delta (Ext. DBD) rule-algorithm of training.
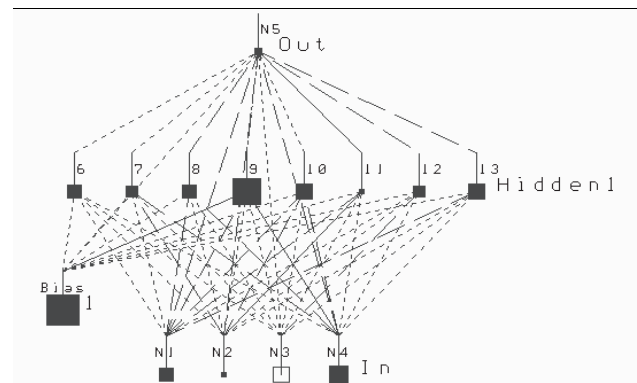

Figure 6 Adopted architecture (4-8-1) BPNN

Analysis of results for adopted BPNN architecture (*RMS* = 3,37%) is given graphically for the training process in Fig. 7.
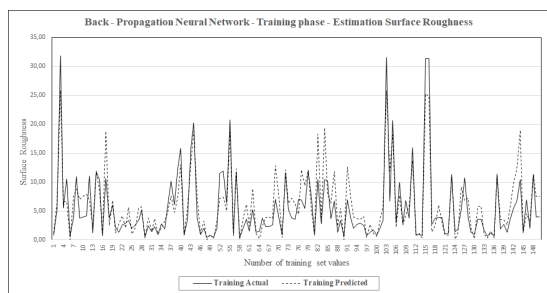
Figure 7 Presentation of achieved BPNN values in the training phase (4 – 8 - 1)

Analysing the data sample (Fig. 2) and the amount of data, in the continuation of experimental work the impact of the amount of data (instances) on the level of *RMS* error will be investigated. The investigation is designed in two steps. In the first step, the data sample is halved and two data sets, identical by the amount of data, are designed. Each data sethad 225 instances in the training phase, while for the testing, 75 instances. For validation purposes, two data sets of 75 instanceseach were also designed. The division of the data sample was made according to the stochastic principle.

For the accepted architecture (4 – 8 - 1) and the realised BPNN characteristics the network was trained again with new data sets, in two steps. In the first step, the training was realised with the first data set (training, testing and validation) and after that with the second data set (division scheme was 225 – 75 - 75 instances). The BPNN network was trained for each data set and the level of *RMS* error (%)was controlled. The results realised in the training phase in the first step were:

- The first data set RMS_I = 4,32%
- The second data set RMS_{II} = 5,78%.
  The *RMS* error range is 1,46%.

In the second step, the datasample was divided in three data sets using the previously described principle. Each data set was modelled according to the established principle: training, testing and validation with the following subset scheme: 150 – 50 - 50 instances. For each data set the BPNN network training process was separately conducted and the level of *RMS* (%) error was controlled. The results realised in the training phase (second step) were the following:

- The first data set RMS_I = 7,09%
- The second data set RMS_{II} = 6,71%
- The third data set RMS_{III} = 6,25%.
  Range of *RMS* error "min tomax" is 0,84%.

From this modelling and researchingof *RMS* error in the function of the amount of data, for investigated and accepted BPNN architecture (4 - 8 - 1) it can be concluded that reducing the amount of data in the training phase results in increasing the *RMS* error.

The remaining experimental work is aimed at modelling a larger number of BPNN architectures with two hidden layers in the form (4 - x - x - 1). The BPNN architecture with adopted and accepted features in previous phases will be investigated. In the results of earlier investigations [23] the architectures with two or three hidden layers did not prove more successful than those with one hidden layer. It has been proven by investigations that nonlinear problems can be very well approximated with one hidden layer.

Tab. 3 shows the realised results of modelling different BPNN architectures in the training process with a whole (complete) data sample of 450 instances and 150 instances in the testing phase.

Table 3 Results of realised BPNN architectures and *RMS* error

NN architecture	RMS / %	NN architecture	RMS / %	NN architecture	RMS / %	NN architecture	RMS / %
4 - 2 - 1 - 1	6,59	4 - 3 - 1 - 1	6,70	4 - 4 - 1 - 1	4,73	4 - 5 - 1 - 1	6,13
4 - 2 - 2 - 1	6,70	4 - 3 - 2 - 1	5,75	4 - 4 - 2 - 1	2,68	4 - 5 - 2 - 1	7,64
4 - 2 - 3 - 1	9,57	4 - 3 - 3 - 1	8,06	4 - 4 - 3 - 1	4,62	4 - 5 - 3 - 1	5,46
4 - 2 - 4 - 1	2,67	4 - 3 - 4 - 1	4,66	4 - 4 - 4 - 1	6,21	4 - 5 - 4 - 1	4,70
4 - 2 - 5 - 1	7,83	4 - 3 - 5 - 1	6,10	4 - 4 - 5 - 1	4,73	4 - 5 - 5 - 1	3,46
4 - 2 - 6 - 1	4,12	4 - 3 - 6 - 1	6,13	4 - 4 - 6 - 1	4,89	4 - 5 - 6 - 1	5,46
4 - 2 - 7 - 1	5,45	4 - 3 - 7 - 1	7,64	4 - 4 - 7 - 1	6,09	4 - 5 - 7 - 1	5,59
4 - 2 - 8 - 1	7,77	4 - 3 - 8 - 1	3,75	4 - 4 - 8 - 1	6,83	4 - 5 - 8 - 1	5,31
4 - 2 - 9 - 1	5,49	4 - 3 - 9 - 1	3,48	4 - 4 - 9 - 1	6,16	4 - 5 - 9 - 1	4,39
4 - 2 - 10 - 1	**2,26**	4 - 3- 10 - 1	2,38	4 - 4 - 10 - 1	6,03	4 - 5 - 10 - 1	4,60
NN architecture	RMS / %	NN architecture	RMS / %	NN architecture	RMS / %	NN architecture	RMS / %
4 - 6 - 1 - 1	3,13	4 - 7 - 1 - 1	5,64	4 - 8 - 1 - 1	4,23	4 - 9 - 1 - 1	4,66
4 - 6 - 2 - 1	4,10	4 - 7 - 2 - 1	5,48	4 - 8 - 2 - 1	6,87	4 - 9 - 2 - 1	7,15
4 - 6 - 3 - 1	5,10	4 - 7 - 3 - 1	5,47	4 - 8 - 3 - 1	3,98	4 - 9 - 3 - 1	2,85
4 - 6 - 4 - 1	5,37	4 - 7 - 4 - 1	5,93	4 - 8 - 4 - 1	6,01	4 - 9 - 4 - 1	5,80
4 - 6 - 5 - 1	5,94	4 - 7 - 5 - 1	5,45	4 - 8 - 5 - 1	4,92	4 - 9 - 5 - 1	4,49
4 - 6 - 6 - 1	2,46	4 - 7 - 6 - 1	2,90	4 - 8 - 6 - 1	5,48	4 - 9 - 6 - 1	3,69
4 - 6 - 7 - 1	3,46	4 - 7 - 7 - 1	7,09	4 - 8 - 7 - 1	2,59	4 - 9 - 7 - 1	6,43
4 - 6 - 8 - 1	6,01	4 - 7 - 8 - 1	3,80	4 - 8 - 8 - 1	4,74	4 - 9 - 8 - 1	4,84
4 - 6 - 9 - 1	3,84	4 - 7 - 9 - 1	6,58	4 - 8 - 9 - 1	4,24	4 - 9 - 9 - 1	3,54
4 - 6 - 10 - 1	7,27	4 - 7 - 10 - 1	6,50	4 - 8 - 10 - 1	6,33	4 - 9 - 10 - 1	4,67

From the results shown in Tab. 3 it can be concluded that in accordance with the least realised *RMS* error the BPNN architecture (4 - 2 - 10 - 1) is selected with two hidden layers and *RMS* error of 2,26% in the training phase (Fig. 8). This BPNN architecture is adopted as the architecture proposed for solving the suggested problem. The process of validation was conducted on the adopted BPNN architecture.
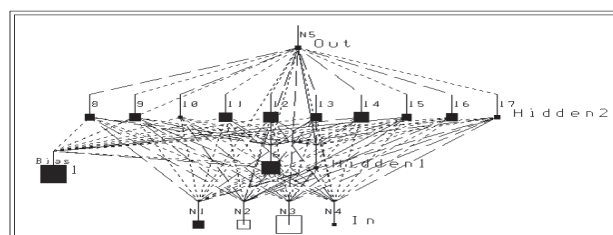

Figure 8 Adopted architecture (4-2-10-1) BPNN

The process of validation is conducted on the new data sample which neural network did not have a chance to use in the training phase and which is an integral part of the results achieved during conducting the experiment (Fig. 2).

In the process of validation the *RMS* error of 4,24% was realised. Display of the achieved results in the training phase is given in Fig. 9, and the results achieved in the validation phase are given in Fig. 10.
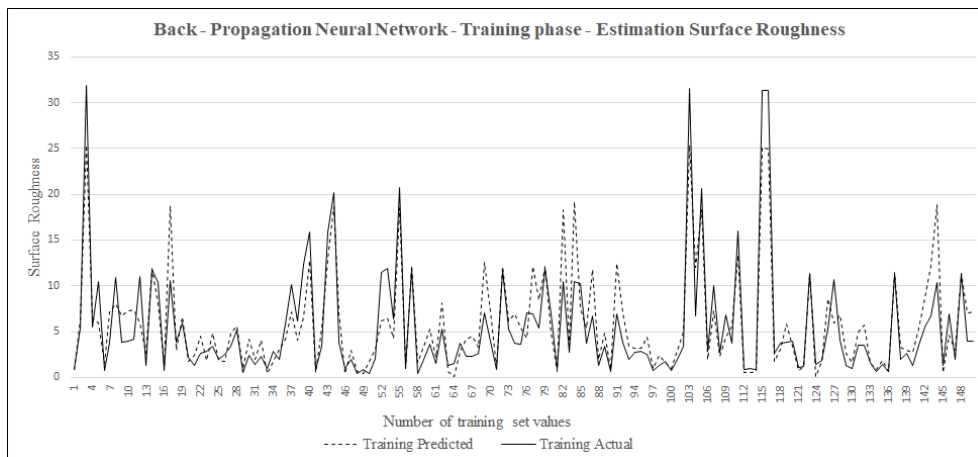

Figure 9 Presentation of realised BPNN (4 - 2 - 10 - 1) values in the training phase
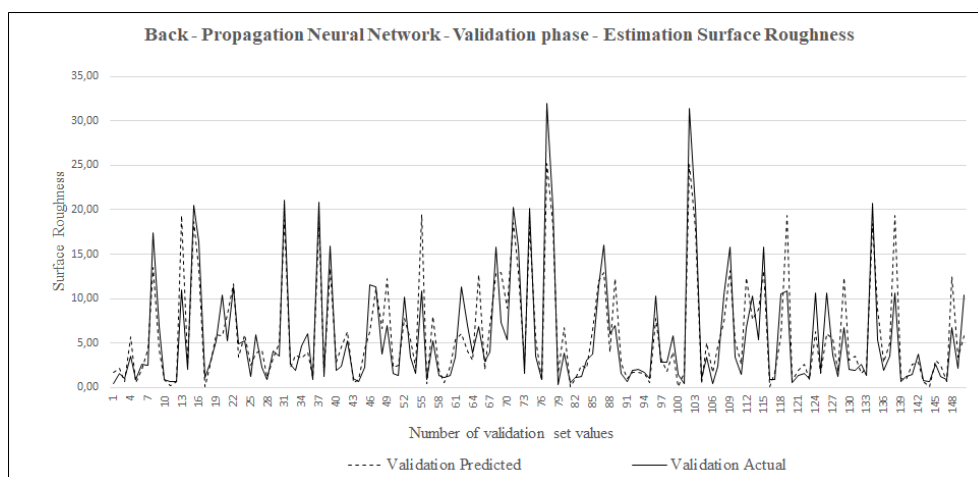

Figure 10 Presentation of realised BPNN (4 - 2 - 10 - 1) values in the validation phase

4 CONCLUSIONS

The investigation carried out in the research of influencing parameters in chip-forming machining process at CNC turning is presented in the paper. In the analysis of the factors of influence on the machining process, cutting speed, feed and depth of cut were selected along with the selected cutting inserts of the tool. The design of experiment defined six cutting inserts and levels of cutting speed (200; 225; 250; 275 and 300 m/min), feed (0,1; 0,2; 0,3; 0,4 and 0,5 mm/rev) and depth of cut (0,5; 1; 1,5; 2 and 2,5 mm) on CNC lathe DMG Moriseiki-CTX 310 Ecoline. The experimental design defined the experimental work conducted on samples of dimension ⌀30 × 350 mm, and the sample material GJS 500 - 7. In accordance with the experimental design, 125 samples were defined with the variations of cutting speed, feed and depth of cut. The complete conducted experimental work was composed of a set of 750 instances. Surface roughness *Ra* measurement was carried out on each sample. The unified data set formed the basis for continuation of experimental work with the BPNN algorithm. The data were prepared before the work on studying acceptable architecture of a neural network began. Various structures were modelled and the

BPNN algorithm parameters varied. Particularly separated and presented were the results of the level of *RMS* error in the function of the hidden layer number of neurons. During the neural network training different training algorithms were also varied which along with the accepted Sigmoid transfer function gave the following results for *RSM*: the Delta rule algorithm 4,82%; the Delta-Bar-Delta rule algorithm 4,73%; the extended Delta-Bar-Delta rule algorithm 3,37% and the Norm. Cum. Delta rule algorithm 4,80%.

The accepted architecture was (4 - 8 - 1) with the Sigmoid transfer function and the algorithm of extended Delta-Bar-Delta rule of the neural network training. After ending and accepting the *RMS* error with one hidden layer, the investigation could continue into the influence of the data set size on *RMS* error. With the accepted BPNN architecture the network training was continued on different amounts of data in two steps. In the first step the data sample was divided in two data sets and the *RMS* errors of 4,32% and 5,78% were realised. In the second step the data sample was divided in three equal data sets. The realised levels of *RMS* error were 7,09%, 6,71% and 6,25%, which leads to the conclusion that by reducing the amount of data the *RMS* error increased for the given

problem of investigation. In continued investigation different BPNN structures were modelled with two hidden layers which generated a smaller error than the structure with one hidden layer. The adopted architecture was the one with two hidden layers (4 - 2 - 10 - 1) and *RMS* error of 2,26% in the training phase, i.e. 4,24% in the validation phase. The adopted BPNN architecture (4 - 2 - 10 - 1) was suggested for solving the problem.

There is a plan for the future research to also investigate the other algorithms of neural networks which successfully solve the problems of prediction.

Acknowledgement

5   REFERENCES

[1] Yang, S. H., Natarajan, U., Sekar, M., & Palani, S. (2010). Prediction of surface roughness in turning operations by computer vision using neural network trained by differential evolution algorithm. *The International Journal of Advanced Manufacturing Technology*, *51*(9-12), 965-971. https://doi.org/10.1007/s00170-010-2668-5

[2] Krishnan, B. R., Vijayan, V., Pillai, T. P., & Sathish, T. (2019). Influence of surface roughness in turning process-an analysis using artificial neural network. *Transactions of the Canadian Society for Mechanical Engineering, 43*(4), 509-514. https://doi.org/10.1139/tcsme-2018-0255

[3] Ulas, H. B. & Ozkan, M. T. (2019). Turning processes investigation of materials austenitic, martensitic and duplex stainless steels and prediction of cutting forces using artificial neural network (ANN) techniques. *Indian Journal of Engineering and Material Sciences, 26*(2), 93-104. Retrieved from: http://nopr.niscair.res.in/handle/123456789/50577

[4] Laouissi, A., Yallese, M. A., Belbah, A., Belhadi, S., & Haddad, A. (2019). Investigation, modelling, and optimization of cutting parameters in turning of gray cast iron using coated and uncoated silicon nitride ceramic tools. Based on ANN, RSM, and GA optimization. *The International Journal of Advanced Manufacturing Technology*, *101*(1-4), 523-548. https://doi.org/10.1007/s00170-018-2931-8

[5] Deshpande, Y. V., Andhare, A. B., & Padole, P. M. (2019). Application of ANN to estimate surface roughness using cutting parameters, force, sound and vibration in turning of Inconel 718. *SN Applied Sciences, 1*(1), 104, 9. https://doi.org/10.1007/s42452-018-0098-4

[6] Jurkovic, Z., Cukor, G., Brezocnik, M., & Brajkovic, T. (2018). A comparison of machine learning methods for cutting parameters prediction in high speed turning process. *Journal of Intelligent Manufacturing, 29*(8), 1683-1693. https://doi.org/10.1007/s10845-016-1206-1

[7] Svalina, I., Simunovic, G., & Simunovic, K. (2013). Machined surface roughness prediction using adaptive neuro fuzzy inference system. *Applied Artificial Intelligence, 27*(9), 803-817. https://doi.org/10.1080/08839514.2013.835233

[8] Meddour, I., Yallese, M. A., Bensouilah, H., Khellaf, A. & Elbah, M. (2018). Prediction of surface roughness and cutting forces using RSM, ANN, and NSGA-II in finish turning of AISI 4140 hardened steel with mixed ceramic tool. *TheInternational Journal of Advanced Manufacturing Technology, 97*(5-8), 1931-1949. https://doi.org/10.1007/s00170-018-2026-6

[9] Dahbi, S., Ezzine, L., & ELMoussami, H. (2017). Modeling of cutting performances in turning process using artificial neural networks. *International Journal of Engineering Business Management, 9*, 1-13. https://doi.org/10.1177/1847979017718988

[10] Du, M., Wang, P., Wang, J., Cheng, Z., & Wang, S. (2019). Intelligent Turning Tool Monitoring with Neural Network Adaptive Learning. *Complexity*, 8431784, 21. https://doi.org/10.1155/2019/8431784

[11] Thangarasu, S. K., Shankar, S., Mohanraj, T., & Devendran, K. (2020). Tool wear prediction in hard turning of EN8 steel using cutting force and surface roughness with artificial neural network. *Proceedings of the Institution of Mechanical Engineers. Part C:Journal of Mechanical Engineering Science, 234*(1), 329-342. https://doi.org/10.1177/0954406219873932

[12] Sofuoglu, M. A., Cakir, F. H., Kushan, M. C., & Orak, S. (2019). Optimization of different non-traditional turning processes using soft computing methods. *Soft Computing, 23*(13), 5213-5231. https://doi.org/10.1007/s00500-018-3471-8

[13] Babu, S. S. & Vinayagam, B. K. (2015). Surface roughness prediction model using adaptive particles warm optimization (APSO) algorithm. *Journal of Intelligent & Fuzzy Systems, 28*(1), 345-360. https://doi.org/10.3233/IFS-141310

[14] Wang, P., Zhang, H., Ye, P., Zhao, T., & Sun, Q. (2017). Research on Surface Roughness Prediction of Turning Parts Based on BP Artificial Neural Network. *Proceedings of the 2nd International Forum on Management, Education and Information Technology Application (IFMEITA 2017)*, Shenzhen, China,133-138. https://doi.org/10.2991/ifmeita-17.2018.23

[15] Šarić, T., Šimunović, G., Šimunović, K., & Svalina, I. (2016). Estimation of Machining Time for CNC Manufacturing using Neural Computing. *International Journal of Simulation Modelling, 15*(4), 663-675. https://doi.org/10.2507/IJSIMM15(4)7.359

[16] Tansel, I. N., Ozcelik, B., Bao, W. Y., Chen, P., Rincon, D., Yang, S. Y., & Yenilmez, A. (2006). Selection of optimal cutting conditions by using GONNS. *International Journal of Machine Tools and Manufacture, 46*(1), 26-35. https://doi.org/10.1016/j.ijmachtools.2005.04.012

[17] Hanafi, I., Cabrera, F. M., Khamlichi, A., Garrido, I. & Manzanares, J. T. (2013). Artificial neural networks backpropagation algorithm for cutting force components predictions. *Mechanics & Industry, 14*(6), 431-439. https://doi.org/10.1051/meca/2013084

[18] Šarić, T., Šimunović, G., & Šimunović, K. (2013). Use of Neural Networks in Prediction and Simulation of Steel Surface Roughness. *International Journal of Simulation Modelling, 12*(4), 225-236. https://doi.org/10.2507/IJSIMM12(4)2.241

[19] Dhas, J. E. R., Stalin, R. S., & Rajeesh, J. (2013). RBF neural network model for machining quality prediction in CNC turning process. *International Journal of Modelling, Identification and Control, 20*(2), 174-180. https://doi.org/10.1504/IJMIC.2013.056190

[20] Pontes, F. J., de Paiva, A. P., Balestrassi, P. P., Ferreira, J. R., & da Silva, M. B. (2012). Optimization of Radial Basis Function neural network employed for prediction of surface roughness in hard turning process using Taguchi's orthogonal arrays. *Expert Systems with Applications, 39*(9), 7776-7787. https://doi.org/10.1016/j.eswa.2012.01.058

[21] Svalina, I., Šimunović, G., Šarić, T., & Lujić, R. (2017). Evolutionary neuro-fuzzy system for surface roughness evaluation. *Applied soft computing, 52*, 593-604.

https://doi.org/10.1016/j.asoc.2016.10.010

[22] Šarić, T., Šimunović, G., Vukelić, Đ., Šimunović, K., & Lujić, R. (2018). Estimation of CNC Grinding Process Parameters Using Different Neural Networks. *Tehnicki vjesnik-TechnicalGazette, 25*(6), 1770-1775. https://doi.org/10.17559/TV-20180419095119

[23] Šarić, T., Šimunović, G., Lujić, R., Šimunović, K., & Antić, A. (2016). Use of soft computing technique for modelling and prediction of CNC grinding process. *Tehnicki vjesnik-Technical Gazette, 23*(4), 1123-1130. https://doi.org/10.17559/TV-20160405151333

Contact information:

Tomislav ŠARIĆ, PhD, Full Professor
(Corresponding author)
Mechanical Engineering Faculty in Slavonski Brod,
University of Slavonski Brod,
Trg Ivane Brlic Mazuranic 2, HR-35000 Slavonski Brod, Croatia
E-mail: tsaric@sfsb.hr

Đorđe VUKELIĆ, PhD, Associate Professor
Faculty of Technical Sciences,
University of Novi Sad,
Trg Dositeja Obradovica 6, 21000 Novi Sad, Serbia
E-mail: vukelic@uns.ac.rs

Katica ŠIMUNOVIĆ, PhD, Full Professor
Mechanical Engineering Faculty in Slavonski Brod,
University of Slavonski Brod,
Trg Ivane Brlic Mazuranic 2, HR-35000 Slavonski Brod, Croatia
E-mail: ksimun@sfsb.hr

Ilija SVALINA, PhD, Assistant Professor
Mechanical Engineering Faculty in Slavonski Brod,
University of Slavonski Brod,
Trg Ivane Brlic Mazuranic 2, HR-35000 Slavonski Brod, Croatia
E-mail: isvalina@sfsb.hr

Branko TADIĆ, PhD, Full Professor
Faculty of Engineering,
University of Kragujevac,
Sestre Janjić 6, 34000 Kragujevac, Serbia
E-mail: btadic@kg.ac.rs

Miljana PRICA, PhD, Full Professor,
Faculty of Technical Sciences,
University of Novi Sad
Trg Dositeja Obradovica 6, 21000 Novi Sad, Serbia
E-mail: miljana@uns.ac.rs

Goran ŠIMUNOVIĆ, PhD, Full Professor
Mechanical Engineering Faculty in Slavonski Brod,
University of Slavonski Brod,
Trg Ivane Brlic Mazuranic 2, HR-35000 Slavonsk iBrod, Croatia
E-mail: gsimun@sfsb.hr