FACTA UNIVERSITATIS
Series: **Mechanical Engineering** Vol. 9, N^o 1, 2011, pp. 9 - 20

PREDICTION OF THE AVERAGE SURFACE ROUGHNESS IN DRY TURNING OF COLD ROLLED ALLOY STEEL BY ARTIFICIAL NEURAL NETWORK

UDC 621.777 669.14 621.906

Velibor Marinković[1]*, Dejan Tanikić[2]

[1]Faculty of Mechanical Engineering Niš, University of Niš, Serbia
[2]Technical Faculty Bor, University of Belgarde, Serbia
E-mail: velmar@masfak.ni.ac.rs

Abstract. *The surface quality of the machined parts is one of the most important product quality characteristics and one of the most frequent customer requirements. In this study, an artificial neural network (ANN) approach for modeling of surface roughness in dry single-point turning of an alloy steel using coated tungsten carbide inserts is presented. The three main cutting parameters consisting of cutting speed, feed rate, and depth of cut are varied in the experiment. Each of the other parameters is treated as constant. The average surface roughness (Ra) is chosen as a measure of surface quality. The data set from major experiment is employed for training a feed-forward three-layer backpropagation ANN. The developed ANN model is tested on the other combinations of the cutting parameters in the given ranges, which are not included in the training process. The results of calculations are in good agreement with the experimental data confirming the effectiveness of ANN approach in modeling of surface roughness in turning process.*

Key words: *Turning Process, Surface Roughness, Artificial Neural Networks*

1. INTRODUCTION

The surface roughness of the machined parts is one of the most significant product quality characteristics. The surface roughness has a great influence on the functional behaviour of the machined parts during exploitation, as well as on the production costs.

The lack of good surface quality cannot satisfy one of the most important technical requirements for mechanical products. On the other side, an extremely high surface quality causes higher production costs and lower overall productivity of cutting operations. In this sense, the desired surface quality is a critical constraint in selecting the optimal cutting parameters in the production process [1], [2], [3], [4].

Received July 16, 2010

* **Acknowledgements**. This work is carried out within the project TR 35034 financially supported by the Ministry of Education and Science of the Republic of Serbia.

The surface finish in turning is found to be influenced in varying amounts by a number of factors, such as cutting speed, feed rate, depth of cut, material characteristics, tool geometry, workpiece deflection, stability and stiffness of the machine tool - cutting tool - workpiece system, built-up edge, cutting fluid, etc.

There are various parameters used to evaluate surface roughness. In the present research for surface finish characterization in turning operations, the average surface roughness (R_a) is selected. It is the most widely used surface finish parameter in industry.

Various methodologies and practices are being employed for the prediction of surface roughness, such as machining theory, classical experimental design, the Taguchi method and artificial intelligence or soft computing techniques. A general review of predicting the surface roughness in machining is available [5].

Karayel [6] developed a three-layer back-propagation neural network for prediction and control of surface roughness in turning process. A large experiment has been carried out on the CNC lathe with various cutting parameters (depth of cut, feed rate and cutting speed). Surface roughness values R_a, R_t, and $R_{\max}$ have been measured as output parameters. The predicted results of the neural network were very close to the measured values.

Risbood et al. [7] investigated the feasibility of using the neural networks for the prediction of surface roughness and dimensional deviation by measuring the cutting force and the radial vibrations of the tool holder. Two different neural networks were created for dry and wet steel turning, as well as for turning by HSS and TiN coated carbide tools.

Özel and Karpat [8] concluded that the improvement of cost effectiveness and productivity can be made to finish hard turning using Cubic Boron Nitride (CBN). In their paper, the experimental data sets for surface roughness and tool flank wear in finish hard turning of various steels and cutting conditions, using CBN tools, were employed to train neural network models. These models are compared to regression models.

Lin et al. [9] simulated surface roughness and cutting force in the turning operation by using the abductive neural network. The best architecture of this neural network can be automatically generated by using the predicted square error (PSE) criterion. To verify the accuracy of the neural network, the second-order mathematical models have been developed for surface roughness and cutting force.

Jiao et al. [10] concluded that a combined neural-fuzzy approach is the most suitable for machining process modeling and control. The fuzzy adaptive network (FAN) was developed for the purpose of surface roughness prediction in turning. The FAN network has both the learning ability and the linguistic representation of the complex phenomenon.

Davim et al. [11] investigated the effects of cutting parameters of on the surface finish in steel turning using the design of the experiment and neural network. For the purpose of experimentation, the authors selected the standard $L_{27}(3^3)$ orthogonal array, based on the Taguchi experimental design. The multiple linear regression and three-layer back-propagation neural network models were developed to study the effect of cutting conditions on surface roughness parameters (R_a and R_t).

In an extensive work [12], various methodologies for the prediction of the surface profile and roughness were presented, and each approach had its advantages and disadvantages. Three major categories are created for classifying these approaches as a pure modeling-based approach, a signal-based approach and an artificial intelligence-based approach. Additionally, the RBF neural network, trained with the adaptive-adjusting parameter, was used for predicting the surface profile.

Samanta et al. [13] presented the surface roughness model using computational intelligence (CI) techniques, which include ANN and adaptive neuro-fuzzy inference system. The results show the effectiveness of CI techniques in modeling surface roughness.

2. Experiment

The parameters (factors) considered in the present paper are cutting speed (*V*), feed rate (*f*) and depth of cut (*a*). Average surface roughness (R_a) was chosen for a target function (response, output).

Since it is obvious that the effects of factors on the selected target function are nonlinear, an experiment with factors at three levels was set up (Table 1). The factor ranges were chosen with different criteria for each factor, aiming at the widest possible range of values, in order to have a better utilization of the proposed models. At the same time, the possibility of the mechanical system and manufacturer's recommendations are taken into account.

Production conditions used in the experiment are shown in Table 2. All of the trials have been conducted on the same machine tool, with the same tool type and the same cutting conditions.

A comprehensive experiment was conducted under these conditions [14]. Part of the experimental results, which refers to the surface finish in a single-point turning process, is analyzed in this study.

A design matrix was constructed on the basis of the selected factors and factor levels (Table 3). The selected design matrix was a full factorial design consisting of 27 rows of coded/natural factors, corresponding to the number of trials. This design provides a uniform distribution of experimental points within the selected experimental hyper-space and the experiment with high resolution.

The choice of this experiment design is derived from the intention to use the same experimental results for different methods of research. In this way, the accuracy, efficiency and practical applicability of each method may be compared.

Average surface roughness values (R_a), shown in Table 3, are the average values of the three measurements.

Table 1. Cutting factors and their levels

Cutting factor	Symbol	Unit	Factor levels		
			Level 1 (Low)	Level 2 (Middle)	Level 3 (High)
Cutting speed	*V*	(m/min)	80	110	140
Feed rate	*f*	(mm/rev)	0.071	0.196	0.321
Depth of cut	*a*	(mm)	0.5	0.125	2.0

Table 2. Machining system, workpiece, measuring equipment

Machine tool	Production lathe PA-C-30 (Potisje-Ada), Three-phase 7.5 kW induction electric motor, Speed range n=20÷2000 rpm, Longitudinal feed rate range f=0.04÷9.16 mm/rev, Max. workpiece diameter d_{max}= 600 mm, Distance from chuck to the tail stock L= 1500 mm
Cutting tool	CNMG 12 04 08 coated tungsten carbide inserts (Sandvik Coromant), PCLNR 32 25 P12 tool holder (Sandvik Coromant)
Workpiece	Č.4732 (42CrMo4- AISI 4140 designation) cold rolled steel; Chemical composition: 1.40% C, 1.00% Cr, 0.20% Mo, 0.90% Mn, 0.25% Si, 0.03% P, 0.10% S; Tensile strength 1050 N/mm^2, Hardness 205 BHN; Workpiece diameter d= 45 mm, Workpiece length l= 250 mm
Cutting fluid	Dry turning
Measuring equipment	Surftest SJ-301 (Mitutoyo) surface profilometer, Cut-off length 0.8 mm; MBS-9 optical microscope

Table 3. Experimental design and results

Trial	Natural factor			Coded factor			Response
	V	f	a	x_1	x_2	x_3	R_a (μm)
1	80	0.071	0.50	1	1	1	3.60
2	80	0.071	1.25	1	1	2	3.61
3	80	0.071	2.00	1	1	3	3.96
4	80	0.196	0.50	1	2	1	4.30
5	80	0.196	1.25	1	2	2	4.955
6	80	0.196	2.00	1	2	3	5.92
7	80	0.321	0.50	1	3	1	5.13
8	80	0.321	1.25	1	3	2	5.28
9	80	0.321	2.00	1	3	3	5.98
10	110	0.071	0.50	2	1	1	2.32
11	110	0.071	1.25	2	1	2	2.745
12	110	0.071	2.00	2	1	3	3.44
13	110	0.196	0.50	2	2	1	2.55
14	110	0.196	1.25	2	2	2	3.405
15	110	0.196	2.00	2	2	3	3.33
16	110	0.321	0.50	2	3	1	3.73
17	110	0.321	1.25	2	3	2	4.005
18	110	0.321	2.00	2	3	3	4.23
19	140	0.071	0.50	3	1	1	1.13
20	140	0.071	1.25	3	1	2	2.79
21	140	0.071	2.00	3	1	3	3.08
22	140	0.196	0.50	3	2	1	1.85
23	140	0.196	1.25	3	2	2	2.835
24	140	0.196	2.00	3	2	3	3.27
25	140	0.321	0.50	3	3	1	3.52
26	140	0.321	1.25	3	3	2	3.605
27	140	0.321	2.00	3	3	3	3.66

3. Artificial Neural Network (ANN)

3.1. Overview of ANN

Inspired by the biological nervous system, the ANNs have found application in many different areas, such as dynamic system identification, complex system modeling, optimization and control, design and classification, speech interpretation, pattern recognition, manufacturing processes simulation, robotics and communication, etc [15].

The ANNs are currently one of the most powerful modeling techniques based on the statistical approach. The ANNs do not require any kind of mathematical model. This technique is especially suitable for modeling a physical phenomenon or systems/processes whose working mechanism is very difficult or even impossible to understand. Thus, calculating with the ANNs is not-algorithmic. They solve a problem by means of learning through examples (data set) rather than by explicit programming based on well-defined rules. The ANNs have some additional advantages, such as the ability to learn incrementally, to remember and generalize, to compute fast and implement easily. The ANN is a suitable tool for working with dispersed data. The data presented to an ANN may be theoretical, experimental or empirical.

The architecture of an ANN consists of at least three layers, i.e. input, hidden and output layers. Any layer consists of a number of units called neurons. The layers are fully interconnected so that each neuron in the preceding layer is connected to all neurons in the succeeding layer. However, no interconnections exist between the neurons in the same layer and the feedback connections in a majority of ANNs. The number of neurons in the input layer (usually several) and the output layer (often only one) depends on the nature of the problem and the goal of the research, i.e. they are defined by the number of selected input (independent) variables and output (dependent) variables.

The number of hidden layers and the number of neurons in each one of them are not defined in advance. They are usually found with a trial-and-error approach, although some heuristic instructions exist even now. Many papers showed that one hidden layer is enough for appropriate model performance. The number of hidden neurons is a critical design parameter. Namely, for a smaller number of hidden layers and neurons, the performance may not be adequate, whereas with too many hidden neurons, the ANN may have poor generalization capability.

An artificial neuron is a simple processor. Since the capability of a single neuron is limited, complex functions can be realized by connecting many neurons. The neurons in the current layers of an ANN receive data (signals) from neurons from the previous layers which, ‘’loaded’’ by multiplication with interconnection weights and biases, make the neuron potential. Interconnection (synaptic) weights and biases are assumed to be small random variables (sometimes with specified distributions). When the neuron potential exceeds a certain threshold, the neuron produces a respective response (signal), which is further forwarded to other neurons of the subsequent layer via interconnections.

The value of the response (signal) at the output of the neuron is obtained by applying an activation function, whose argument is the neuron’s potential. The activation (transfer) functions are needed for the sake of introducing non-linearity into the ANN, which makes multi-layered ANNs so powerful. Most frequently, a linear function is applied at the output layer of the ANN and the sigmoid (s-shaped) functions in the hidden layers (because of their non-linearity and differentiability). The following relations mathematically formulate the above text:

$$a_i = \sum_{j=1}^{N_x} w_{ij} \cdot x_j - \theta_i \tag{1a}$$

$$y_i = f(a_i) \tag{1b}$$

where a_i is the potential of i-th neuron, w_{ij} is the adjustable interconnection weight, between j-th neuron in the previous layer and i-th neuron in the current layer, x_j is the output from j-th neuron also input into i-th neuron, θ_i is the threshold of activation of i-th neuron, N_x is the total number of inputs into i-th neuron, y_i is the output from i-th neuron, and f is the activation function (transfer function).

Once the architecture of ANN is determined, the ANN has to be trained on the examples (training data set) using the selected training algorithm, where each example is a pair of input and output data. The training process is initialized by assigning random weights and biases to all the interconnections. Training (learning) involves presenting the examples to the ANN at input layer, calculating the error at output layer, and then adjusting the interconnection weights and biases by small increments to reduce the overall error. The input layer receives information (signals) from the external source and transmits this information to the ANN for processing.

The training is an iterative process, which is repeated until the ANN is stabilized or until error function (mean squared error (MSE), sum squared error (SSE)) between the predicted values and the corresponding desired (target) values is reduced below a previously defined threshold. It is known that as the number of iterations in the training process increases, the training error falls, but the generalization error rises. Such an ANN is said to be "overfitting’’. Overfitting is usually an indication that either the ANN used is too large or the training data set is too small.

The training process is the most significant part of ANN modeling. There are many techniques for training an ANN. The most commonly used is the back-propagation (BP) training procedure. The BP ANN is designed to operate as a multilayer fully-interconnected feed-forward ANN, with a particular training algorithm for supervised learning. The BP algorithm, also called the generalized delta rule, is essentially a gradient descent search technique. In order to minimize the networks’ error function, the BP algorithm adjusts the interconnection weights by taking small steps, iteratively, along the direction of the negative gradient. There are many different standard training algorithms for a BP ANN. One of the most widely used is the Levenberg-Marquardt (L-M) algorithm, which offers a good generalization and a fast convergence in less iteration, especially for smaller and moderate large ANNs.

For the standard gradient descent algorithm, the total sum of the squared error in the output layer is determined from the following relation [16]:

$$E_T = \sum_{k=1}^{N} E(k) = \sum_{k=1}^{N} \frac{1}{2} \sum_{j=1}^{N_o} [y_j(k) - y_j^t(k)]^2 \tag{2}$$

where $E(k)$ is the sum of the squared error for k-th sample from a set of samples (input-output data) for training, N is the total number of samples for training, N_0 is the number of output neurons, $y_j(k)$ is the current output of j-th neuron for k-th sample, and $y^t_j(k)$ is the target output j-th neuron for k-th sample. The process of computing error $E(k)$ in Eq. (2) is called a forward pass.

In the general case, for the ANN with back-propagation, regardless of the training procedure, the following equation is valid (in vector notation) for adjusting interconnection weights [16]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \eta \frac{\partial \mathbf{E}}{\partial \mathbf{w}} + \alpha \Delta \mathbf{w}(k+1) \tag{3}$$

where $\mathbf{w}(k)$ is the vector of interconnection weights in current (k-th) iteration, $\mathbf{w}(k+1)$ is the vector of interconnection weights in sub-sequent iteration, $\Delta\mathbf{w}(k+1)$ is the increment of vectors of interconnection weights in the previous operation, η is the learning rate, and α is the momentum term. The process of computing partial derivatives $\partial\mathbf{E}/\partial\mathbf{w}$ in Eq. (3) is called a backward pass.

The training process can be interrupted even when the desired error threshold is not achieved. Namely, one ANN may have adequate performances even when training error and generalization error do not exceed the acceptable levels. The other method to stop the training implies constraining the number of training iterations.

After the training, the ANN may be tested in order to evaluate its predictive and generalization performances. Testing the ANN is carried out by applying a new input data set, which was not included in the training process.

It is widely reported that the architecture of an ANN, distribution and rang of the training data, normalization of input-output pairs, appropriate selection of activation functions and the training algorithm have a significant influence on the effectiveness and performance of the trained ANN.

3.2. ANN Implementation

The inputs of ANN were cutting speed (V), feed rate (f) and depth of cut (a). The output was average surface roughness (R_a).

The experimental data given in Table 3 were used for ANN training. The number of input-output data is relatively small. However, many studies have shown that a comprehensive set of data is not necessary for the successful training of an ANN [7], [17], [18], [19]. Truly, literature contains cases of application of unnecessarily large and complex ANNs in terms of their architecture. In this sense, the proper choice of type, architecture, training algorithm and other ANN parameters are more important, in addition to the statistical properties of the data collected.

The powerful L-M algorithm, which belongs to the type of classical BP algorithm, was used for ANN training. The non-linear hyperbolic tangent was selected for the transfer function of the hidden layer, while linear transfer function was selected for the neuron in the output layer. The mean squared error (MSE) represents the optimization criterion for interconnection weights. The training data set and the testing data set are presented to the ANN by the batch method.

The initial neural network architecture $3 - N_h - 1$ was selected. The number of neurons in the hidden layer is defined according to the number of input-output pars for training ANN and the number of interconnection weights and biases. In this specific case, the following inequality is valid: $N_h < 6$. In order to determine the optimal architecture, the trial-and-error approach is carried out by adjusting the number of neurons in the hidden layer. Five different ANNs with a various number of neurons in the hidden layer (3-1-1, 3-2-1, 3-3-1, 3-4-1, and 3-5-1) have been created and tested.

Since the result of the training greatly depends on the initialization of the interconnection weights, each ANN was trained ten times. Both the number of iterations and the error criterion were considered together. The number of iterations was limited to 100 in order to prevent an overfitting effect. After the training of all ANNs with the same other parameters, the ANN with the highest accuracy, i.e. with the highest generalization ability, was chosen.

Considering all afore said, in this specific task, the optimum architecture 3-5-1 of a three-layer BP ANN was selected, and its performance is shown in Fig. 1. Additional trials were used to test this ANN (No 28÷34, Table 4), where they kept previously adopted ranges of design factors. Consequently, three-fourths of the whole experimental data set has been employed for training and one-fourth of the data set has been used for testing the trained ANN.

The MATLAB version 7.0.1 (R14) software package was used to create, train, and test the ANNs.

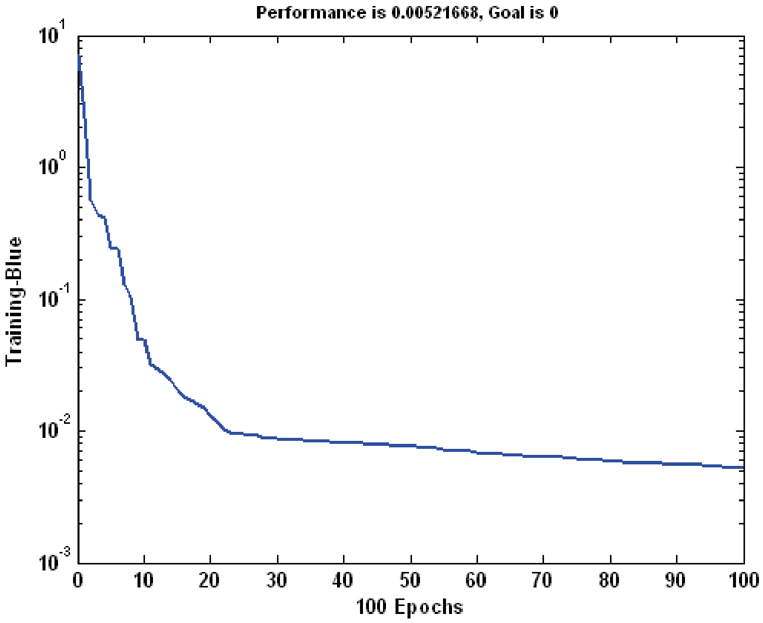

Fig. 1. Mean squared error (MSE) versus the number of training iterations

Table 4. Experimental and predicted results and absolute percentage error

Trial	Natural factor			Coded factor			Response		Error
	V	f	a	x_1	x_2	x_3	R_a	$\hat{R}_a$	$\|\delta_i\|$ (%)
1	80	0.071	0.50	1	1	1	3.60	3.5963	0.1039
2	80	0.071	1.25	1	1	2	3.61	3.6287	0.5185
3	80	0.071	2.00	1	1	3	3.96	3.9464	0.3437
4	80	0.196	0.50	1	2	1	4.30	4.3257	0.5984
5	80	0.196	1.25	1	2	2	4.955	5.1135	3.1980
6	80	0.196	2.00	1	2	3	5.92	5.7438	2.9768
7	80	0.321	0.50	1	3	1	5.13	4.9093	4.3013
8	80	0.321	1.25	1	3	2	5.28	5.4317	2.8729
9	80	0.321	2.00	1	3	3	5.98	6.0156	0.5954
10	110	0.071	0.50	2	1	1	2.32	2.3134	0.2828
11	110	0.071	1.25	2	1	2	2.745	2.7739	1.0511
12	110	0.071	2.00	2	1	3	3.44	3.4136	0.7674
13	110	0.196	0.50	2	2	1	2.55	2.5425	0.2958
14	110	0.196	1.25	2	2	2	3.405	3.3946	0.3062
15	110	0.196	2.00	2	2	3	3.33	3.3634	1.0034
16	110	0.321	0.50	2	3	1	3.73	3.7369	0.1856
17	110	0.321	1.25	2	3	2	4.005	4.0182	0.3285
18	110	0.321	2.00	2	3	3	4.23	4.2375	0.1765
19	140	0.071	0.50	3	1	1	1.13	1.1283	0.1545
20	140	0.071	1.25	3	1	2	2.79	2.7571	1.1782
21	140	0.071	2.00	3	1	3	3.08	3.0675	0.4068
22	140	0.196	0.50	3	2	1	1.85	1.8499	0.0040
23	140	0.196	1.25	3	2	2	2.835	2.8238	0.3968
24	140	0.196	2.00	3	2	3	3.27	3.2396	0.9295
25	140	0.321	0.50	3	3	1	3.52	3.4992	0.5899
26	140	0.321	1.25	3	3	2	3.605	3.6289	0.6618
27	140	0.321	2.00	3	3	3	3.66	3.7239	1.7461
28	140	0.085	0.50	3	1.112	1	1.14	1.2323	8.0960
29	140	0.150	0.50	3	1.632	1	1.57	1.7383	10.7218
30	110	0.150	1.00	2	1.632	1.666	2.93	2.7353	6.6445
31	90	0.098	1.50	1.333	1.216	2.333	3.30	3.6331	10.0937
32	130	0.098	1.50	2.666	1.216	2.333	3.08	2.8280	8.1829
33	130	0.071	1.00	2.666	1	1.666	2.18	2.3914	9.6975
34	130	0.321	0.50	2.666	3	1	3.53	3.4427	2.4735
The mean absolute percentage error $\|\bar{\delta}\|$= 2.41 (%)									

After the training, the analysis of ANN responses to the presented data set was performed. The responses of the trained ANN, for data sets presented in the training and testing phase, are shown in Fig.2. For an ideal prediction, all points should lie on the line passing through the origin and inclined at 45^0. It is seen here that most of the points are close to this line. Hence, this model provides for reliable prediction.

Also, the criterion used to estimate the efficiency and ability of the ANN model to predict average surface roughness may be absolute percentage error ($|\delta_i|$) which is defined by Eq. (4):

$$|\delta_i| = \left|\frac{\hat{R}_{ai} - R_{ai}}{R_{ai}}\right| 100\ (\%) \tag{4}$$

where $\hat{R}_{ai}$ and R_{ai} represent the predicted average surface roughness and measured average surface roughness for i-th trial, respectively.

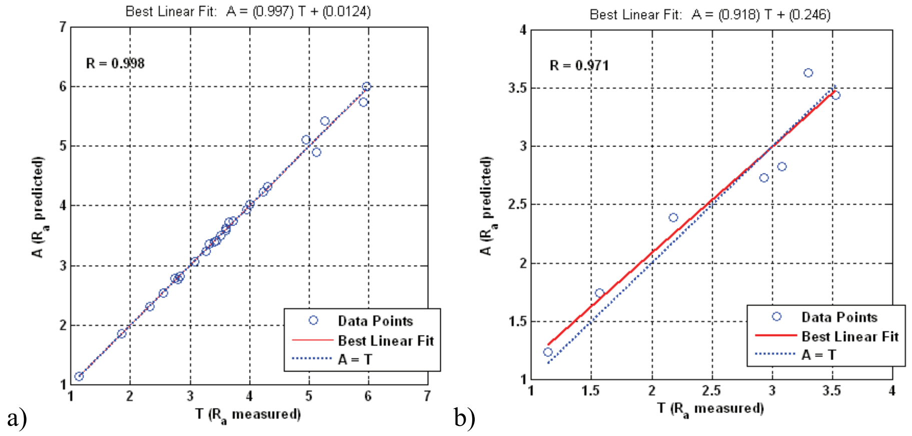

Fig. 2. Correlation between prediction of proposed ANN model and experiment for surface roughness: a) training of the ANN, b) testing of the ANN

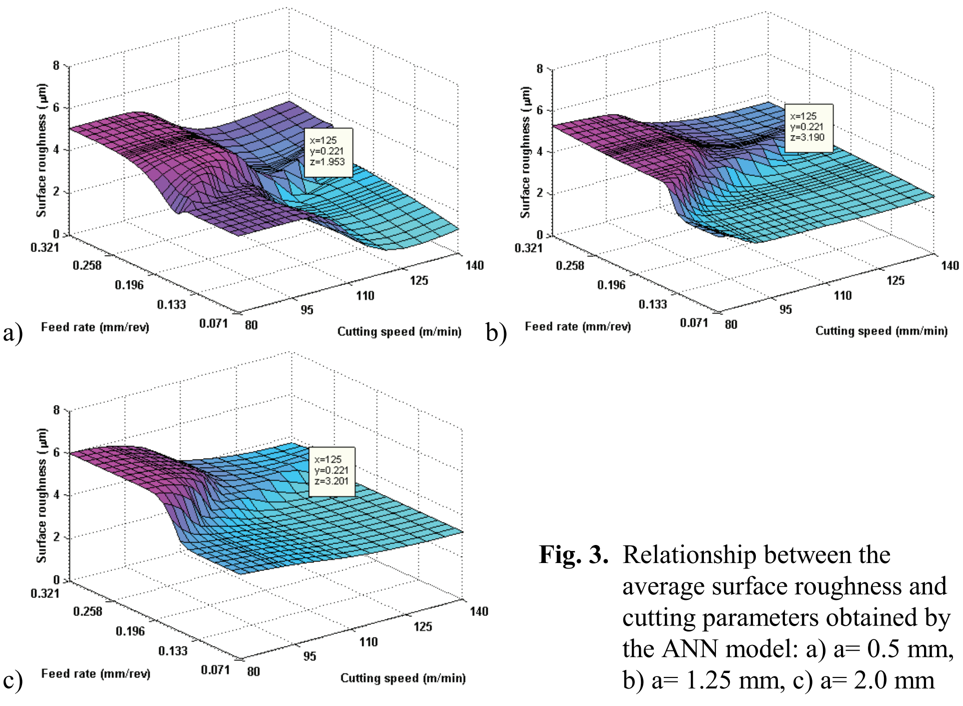

Fig. 3. Relationship between the average surface roughness and cutting parameters obtained by the ANN model: a) a= 0.5 mm, b) a= 1.25 mm, c) a= 2.0 mm

The errors of the selected ANN model according to the equation (4) are given in Table 4. The calculation results show that the proposed ANN surface roughness model has a high accuracy rate for predicting the surface roughness for various combinations of cutting conditions. Fig. 3 presents the capability of the ANN model to approximate experimental results for the given cutting conditions.

3. CONCLUSION

This paper presents research of various cutting parameters affecting the surface roughness in dry single-point turning of alloy steel using the coated tungsten carbide inserts. Modeling of surface roughness is conducted by applying the three-layer BP ANN. The main experiment is carried out according to plan-matrix of full factorial design of type 3^3. Also, one separate experiment is added in order to test the trained ANN and verify its generalization ability.

Very good performances of the proposed ANN model are achieved, although the training process is performed on a limited number of input/output data.

On the basis of the obtained results, it can be concluded that the cutting speed has the most dominant effect on the observed surface roughness, followed by the feed rate and depth of cut, whose influences on surface roughness are at the middle level. The surface roughness tends to decline along with increasing cutting speed and decreasing feed rate and depth of cut.

The proposed artificial neural network model is found to be capable for the prediction of surface roughness within a reasonable degree of accuracy.

Further research should be focused on extending this study by using other cutting conditions in order to gain a better understanding of the surface roughness forming mechanism in the turning process.

REFERENCES

1. Jacobs, H.J., Jacob, E., Kochan, D., 1972, *Spanungsoptimierung*. VEB Verlag Technik, Berlin, 279 p. (in German).
2. Tolouei-Rad, M., Bidhendi, I.M., 2001, *On the optimization of machining parameters for milling operation,* Journal of Material Processing Technology, 108, pp.286-293.
3. Al-Ahmari, A.M.A., 2001, Mathematical model for determining parameters in multipass turning operations with constraints, International Journal of Production Research, 39, pp. 3367-3376.
4. Marinković, V., 2004, *Automated choice of the optimal cutting regime in the peripheral milling process*, Proc. 4th International Conference - RaDMI 2004, Zlatibor, pp. 221-227.
5. Benardos, P.G., Vosniakos, G.-C., 2003, *Predicting surface roughness in machining: a review*, International Journal of Machine Tools and Manufacture, 43, pp. 833-844.
6. Karayel, D., 2009, Prediction and control of surface roughness in CNC lathe using artificial neural network, Journal of Material Processing Technology, 209, pp. 3125-3137.
7. Risbood, K.A., Dixit, U.S., Sahasrabudhe, A.D., 2003, *Prediction of surface roughness and dimensional deviation by measuring cutting force and vibration in turning process*, Journal of Material Processing Technology, 132, pp. 203-214.
8. Özel, T., Karpat, Y., 2005, Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks, International Journal of Machine Tools and Manufacture, 45, pp. 467-479.
9. Lin, W.S., Lee B.Y., Wu, C.L., 2001, *Modeling the surface roughness and cutting force for turning,* Journal of Material Processing Technology, 108, pp. 286-293.

10. Jiao, Y. Shuting L., Pei Z.J., Lee E.S., 2004, *Fuzzy adaptive networks in machining process modeling: surface roughness prediction for turning operations*, International Journal of Manufacturing Research, 44, pp. 1643-1651.
11. Davim, J.P., Gaitonde, V.N., Karnik S.R., 2009, Investigations into the effect of cutting conditions on surface roughness in turning of free machining steel by ANN models, Journal of Material Processing Technology, 205, pp. 16-23.
12. Lu, C., 2008, *Study on prediction of surface quality in machining process*, Journal of Material Processing Technology, 205, pp. 439-450.
13. Samanta, B., Erevelles, W., Omurtag, Y., 2009, *Prediction of workpiece surface roughness using soft computing*, Proc. ImechE, Part B: Journal of Engineering Manufacture, 222, pp. 1221-1232.
14. Tanikić, D., 2009, Modeling of the correlations among metal cutting process parameters using the adaptive neuro-fuzzy systems. PhD Thesis, Faculty of Mechanical Engineering, University of Niš, 172 p.
15. Dreyfus, G., 2005, *Neural Networks*, Springer Verlag, Berlin/ Heidelberg, 497 p.
16. Bose, N. K., Liang, P., 1996, Neural Network Fundamentals with Graphs, Alogorithms and Applications, Mc Graw-Hill, New York, 478 p.
17. Altinkok, N., Koker, R., Neural network approach to prediction of bending strength and hardening behaviour of particulate reinforced (Al-Si-Mg)-aluminium matrix composites, Materials and Design, 25, pp. 595-602.
18. Tsao C.C., Hocheng H., 2008, Evaluation of thrust force and surface roughness in drilling composite material using Taguchi analysis and neural network, Journal of Material Processing Technology, 203, pp. 342-348.
19. Marinković, V., 2009, Application of artificial neural network for modeling the flash land dimensions in the forging dies, Journal of Mechanical Engineering, 55(1), pp. 64-75.

PREDIKCIJA HRAPAVOSTI POVRŠINE U PROCESU SUVOG STRUGANJA HLADNO VALJANOG LEGIRANOG ČELIKA POMOĆU VEŠTAČKE NEURONSKE MREŽE

Velibor Marinković, Dejan Tanikić

Kvalitet površine obrađenih delova je jedan od najvažnijih karaktertistika kvaliteta proizvoda i jedan od najčešćih zahteva kupca. U ovoj studiji je prikazana primena veštačke neuronske mreže (VNM) u modelovanju hrapavosti površine pri suvom struganju legiranog čelika presvučenim pločicama od tvrdog metala. U eksperimentu su varirana tri glavna parametra: brzina rezanja, korak i dubina rezanja. Svi ostali parametric su tretirani kao konstante. Kao merilo kvaliteta površine izabrano je srednje aritmetičko odstupanje profila (Ra). Podaci glavnog eksperimenta su upotrebljeni za treniranje direktne troslojne VNM sa propagacijom greške unazad. Razvijeni VNM model je bio testiran na drugim kombinacijama parametara rezanja u datim intervalima variranja, koji nisu bili uključeni u proces treniranja. Rezultati proračuna su u dobroj saglasnosti sa eksperimentalnim podacima i potvrđuju efikasnost VNM u modelovanju hrapavosti površine u procesu struganja.

Ključne reči: *struganje, hrapavost površine, veštačke neuronske mreže*