Neural network classification of surface quality after hard turning of 105WCr6 steel

D A Rastorguev and A A Sevastyanov

Togliatti State University, Belarusian Street, 14, Togliatti City, 445020, Samara Region, Russian Federation

E-mail: alex-119977@yandex.ru

Abstract. The paper presents the results of a surface quality study after hard turning on a CNC lathe. Ring workpieces made of 105WCr6 steel and hardened to HRC 55 are used in this work. Data was obtained on surface quality and type of chips in a three-factor experiment for end face cutting. In order to assess the surface quality, it was photographed on an optical microscope with 4, 10, 40 times magnification. The surface quality was evaluated by traces of processing and divided into three types: the absence of moire, a clear moire, and an intermediate type of surface. The chip morphology was divided into the following categories: discontinuous, snarled and ribbon chips. To predict both parameters for different cutting conditions artificial neural networks (ANNs) were used. Different ANNs are applied to achieve the best classification results. In this work probabilistic neural network (PNN), feedforward network and learning vector quantization (LVQ) network are used. The results of modeling all networks are similar and can be used for technological preparation of production.

1. Introduction

Machining of hard metals is an important technological problem nowadays. One of the most promising methods of finishing metals is hard turning which can replace grinding in the nearest future. To reduce the cost of an expensive tool in hard turning, studies were conducted to determine the effect of the cutting mode on the roughness, tool wear and cutting force [1]. The dependence between the input and output parameters is formed using a multilayer feed-forward ANN. A similar problem is solved in the article [2], which compares regression analysis and neural networks, which showed the best results. In addition to the insert material, as in the previous study, the geometry of the cutting tool can also be taken into account [3].

ANNs have become an efficient mathematical method for solving many difficult problems like data classification in technological research and widely used for modeling processing such as turning. The more advanced modeling techniques using a combination of ANNs and a genetic algorithm for multi-objective optimization in turning operations are presented in [4]. In the article [5] several architectures of network were applied to predict the cutting force depending on the processing parameters. The adaptive neuro-fuzzy inference system can be used for prediction the surface roughness and cutting temperature [6]. Mainly in studies the approximating feed-forward ANNs trained using error back-propagation training algorithm were used.

The main task of this work is to solve the problem of prediction of surface quality after hard turning of 105WCr6 steel with high reliability.
2. Actuality

Hard turning is one of the best methods of machining hardened materials due to its economical, technological and ecological advantages. The roughness is formed during hard turning by kinematic processes as a joint action of feeding, cutting tool geometry and by thermomechanical processes in the cutting zone, which result in the form of various types of chips. With increasing cutting speed, chatter is amplified at frequencies corresponding to the natural frequencies of oscillation of the cutter, which affects the surface roughness. Therefore, roughness prediction is a difficult task for which ANNs are well suited. ANN has the advantage of generalizing the modeling results with high precision solutions. The actual problem is to develop the theory of hard turning to be able to predict the quality of this processing with high accuracy.

3. Experimental technique

The study’s purpose is to determine the range of acceptable cutting conditions for hardened 105WCr6 steel to ensure high quality of the machined surface. This steel is used for the manufacture of cutting and measuring tools with high requirements in terms of accuracy, roughness, dimensional stability.

Experimental studies were performed on a lathe model 16B16T1C1 with a FlexNC CNC system [7]. The workpieces were rings of steel 105WCr6 with a hardness of 55 HRC with geometrical parameters: outer diameter 55 mm, inner diameter 30 mm, thickness 20 mm. No coolant was used during processing. End face cutting of the samples was performed using PCLNR 2525M 12 plate NP-CNGA120404GA2 Mitsubishi and the plate material was cubic boron nitride (CBN), in the following modes: cutting speed - 150, 250, 300, 400 m / min; feed - 0.1; 0.2; 0.3 mm / rev; depth of cut - 0.1; 0.2; 0.3; 0.4 mm.

Photos of the machined samples were taken with magnification of x4, x10 and x40 using a «Labomet» optical microscope. Photos of chips also were obtained and systematized by types in three groups: discontinuous, snarled and ribbon chips.

4. Research results

Correlation between cutting modes and surface quality after hard turning was investigated in the experiment from 48 tests. Quality of surface and the nature of chip formation were evaluated by using photos. Examples of different types of surface obtained in the photos are shown in the figures 1, 2. Summarized results of the experiment are shown in table 1. The main task is to obtain a surface quality prediction technique based on the method of neural network design.

5. ANN classification

ANNs are widely used for such tasks as classification, clustering and pattern recognition problems and also for modeling and forecasting of various technological processes parameters [8]. Using ANN to form technological modes in hard turning has not been applied. In this research probabilistic neural network (PNN), learning vector quantization (LVQ) network and feedforward classification network are used to predict the type of the machining surface.

To employ ANN with the experimental data it is necessary to make next steps. The input factors (cutting speed, depth of cut, feed) are selected first to form input vector \( \mathbf{x} \) with dimension \( k = 3 \). The output vector \( \mathbf{y} \) with appropriate classes of surface quality is formed. The source data for learning is prepared. The dataset must be randomized, normalized, and divided on training, testing and validation groups. Source data of 48 experiments are divided on three groups: training, validating and testing in proportion 75–15–10 %. Validation is necessary for exclusion of overlearning of neural network. Testing is needed to estimate the efficiency of neural network work after training.

Source data normalization is necessary to reduce the data to a single scale and it is performed as:

\[
x_i = \frac{x_i - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}},
\]

where \( X_i \) is actual value of normalized parameter, \( X_{\text{min}}, X_{\text{max}} \) are minimum and maximum value of the parameter, \( x_i \) is parameter value after normalization.
Table 1. Experimental results.

| Sample № | v, m/min | s, mm/rev | t, mm | Surface quality (Chip morphology) | Sample № | v, m/min | s, mm/rev | t, mm | Surface quality (Chip morphology) |
|----------|-----------|------------|-------|-----------------------------------|----------|-----------|------------|-------|-----------------------------------|
| 1        | 150       | 0.1        | 0.1   | 1 (2)                             | 25       | 150       | 0.2        | 0.1   | 1 (3)                             |
| 2        | 150       | 0.1        | 0.2   | 1 (2)                             | 26       | 150       | 0.2        | 0.2   | 2 (2)                             |
| 3        | 150       | 0.1        | 0.3   | 1 (3)                             | 27       | 150       | 0.2        | 0.3   | 2 (1)                             |
| 4        | 250       | 0.1        | 0.1   | 1 (3)                             | 28       | 150       | 0.2        | 0.4   | 3 (1)                             |
| 5        | 250       | 0.1        | 0.2   | 1 (2)                             | 29       | 250       | 0.2        | 0.1   | 2 (2)                             |
| 6        | 250       | 0.1        | 0.3   | 1 (2)                             | 30       | 250       | 0.2        | 0.2   | 3 (2)                             |
| 7        | 250       | 0.1        | 0.4   | 2 (2)                             | 31       | 250       | 0.2        | 0.3   | 3 (1)                             |
| 8        | 250       | 0.3        | 0.1   | 3 (3)                             | 32       | 250       | 0.2        | 0.4   | 3 (1)                             |
| 9        | 250       | 0.3        | 0.2   | 3 (1)                             | 33       | 300       | 0.2        | 0.1   | 2 (2)                             |
| 10       | 250       | 0.3        | 0.3   | 3 (1)                             | 34       | 300       | 0.2        | 0.2   | 3 (2)                             |
| 11       | 250       | 0.3        | 0.4   | 3 (1)                             | 35       | 300       | 0.2        | 0.3   | 3 (1)                             |
| 12       | 300       | 0.1        | 0.1   | 1 (3)                             | 36       | 300       | 0.2        | 0.4   | 3 (1)                             |
| 13       | 300       | 0.1        | 0.2   | 1 (2)                             | 37       | 400       | 0.1        | 0.1   | 1 (2)                             |
| 14       | 300       | 0.1        | 0.3   | 1 (1)                             | 38       | 400       | 0.1        | 0.2   | 1 (2)                             |
| 15       | 300       | 0.1        | 0.4   | 2 (1)                             | 39       | 400       | 0.1        | 0.3   | 2 (2)                             |
| 16       | 300       | 0.3        | 0.1   | 3 (1)                             | 40       | 400       | 0.1        | 0.4   | 2 (1)                             |
| 17       | 300       | 0.3        | 0.2   | 3 (1)                             | 41       | 400       | 0.3        | 0.1   | 3 (1)                             |
| 18       | 300       | 0.3        | 0.3   | 3 (1)                             | 42       | 400       | 0.3        | 0.2   | 3 (1)                             |
| 19       | 300       | 0.3        | 0.4   | 3 (1)                             | 43       | 400       | 0.3        | 0.3   | 3 (1)                             |
| 20       | 150       | 0.3        | 0.1   | 3 (3)                             | 44       | 400       | 0.3        | 0.4   | 3 (1)                             |
| 21       | 150       | 0.3        | 0.2   | 3 (3)                             | 45       | 400       | 0.2        | 0.1   | 3 (2)                             |
| 22       | 150       | 0.3        | 0.3   | 3 (2)                             | 46       | 400       | 0.2        | 0.2   | 3 (2)                             |
| 23       | 150       | 0.3        | 0.4   | 3 (1)                             | 47       | 400       | 0.2        | 0.3   | 3 (1)                             |
| 24       | 150       | 0.1        | 0.4   | 1 (3)                             | 48       | 400       | 0.2        | 0.4   | 3 (1)                             |

Surface quality: 1 – «clean» surface, 2 – intermediate quality, 3 – surface with moire.
Chip morphology: Discontinuous chips (1), Snarled chips (2), Ribbon chips (3).
Then the type and the architecture of ANN must be selected. It means that number of layers in the network and the number of neurons in each layer should be set up. The input layer contains as many neurons as there are selected input parameters. The number of neurons in the hidden layer may be varied from 2 to the number of training data. For the chosen number of neurons in a hidden layer the parameter of learning speed $\mu$ must be defined. The next step is to choose the learning algorithm and to teach the network to define weights and biases. The common feature of these networks is the supervised learning algorithm [9]. After that, the network must be tested.

If the error value in the testing is more than doubled root mean square error of learning (RMS) or more than defined value then relevant data set having maximum error is shifted to training set with adding new test data instead of displaced and ANN training must be repeated. This cycle repeats until the network reaches defined accuracy. RMS is given as:

$$\text{RMS}_{\text{err}} = \sqrt{\frac{\sum_{i=1}^{d}(Y_i - y_i)^2}{d}},$$  \hspace{1cm} (2)

where $Y_i$ is the desired result (obtained from experimental data), $y_i$ is the predicted result from modeling the network, $d$ is a number of tests in experiment.

Figures 3-5 shows architectures of neural networks, which used for solving the classification problem. All three network are clustering the input data into groups according to the classes of roughness and type of chip using different algorithms.

PNN while estimating probabilities of belonging an input element to a defined class by choosing the most probable alternative. This kind of network has advantages over feedforward networks because in this case less time for learning is required. And the disadvantages of such network are large memory usage and possible low speed of work. In the hidden layer 36 neurons exist according to the number of input vectors for learning. These neurons have a sort of radial-basis activation function called Gaussian function (on the scheme called «radbas») of following form:

$$a = e^{-2^n},$$  \hspace{1cm} (3)

where

$$n = \|x - WI\| / b,$$  \hspace{1cm} (4)

where $b$ is the bias vector, $WI$ is the weight vector of the hidden layer.

$$n = \|\text{dist}\| = \sqrt{(x_{k} - WI_{k})^2} - b_{k}$$  \hspace{1cm} (5)

is the Euclidian distance between centers of radial functions obtained from the training input vector (weights) $WI$ and the input vector $x$. The distance estimates proximity of input vector $x$ and weights of each neuron. After the signal passes through the activation function those from neurons in hidden layer whose weights much differ from input vector $x$ will be almost equal 0. In hidden layer the neuron with closest weights to input vector gives value nearly equal 1. In the output layer there are three neurons according to the number of classes with competitive activation («competo») function. This function is expressed as:

$$d_{k} = \begin{cases} 1, k = k^*; k^* = \arg(\max n_{k}) \\ 0, k \neq k^*. \end{cases}$$  \hspace{1cm} (6)

The output layer weight vector $WL$ corresponds to the target vector $T$ got from output vector $Y$. Each vector of target classes $T$ has a 1 only in the row associated with that particular class of input, and 0’s elsewhere. In modeling the competitive function for the largest vector $n_{2k}$ forms an output signal equal 1 and 0 for the others. A compete transfer function of the output layer picks the maximum of the hidden layer sums as probability of contributions for each class of inputs.
Architectures of probabilistic neural network (PNN).

Architectures of two-layer feedforward network for pattern recognition.

Figure 3. Architectures of probabilistic neural network (PNN).

Figure 5. Architectures of two-layer feedforward network for pattern recognition.

Description of neural network elements: $n_{ik}$ is an input of the $k$-th neuron of the $i$-th layer; $a_i$ is the $i$-th layer output after activation; $x$ is an input layer vector; $y$ is a neural network output; $i$ is a number of layer, $i = 1, 2$; $W_L$ is the weight vector of the output layer; $W_I$ is the weight vector of the hidden layer; $b$ is the bias vector of the hidden layer; $N_i$ is the number of neurons in the $i$-th layer; $\|\text{dist}\|$ is Euclidian distance between centers of radial functions and the input vector $x$.

As training input vector $P$ is used in form of an array with size 3x36. This array includes data for 36 samples with next parameters: cutting speed on the outer diameter $v = (150; 250; 300; 400)$ m/min, feed $s = (0.1; 0.2; 0.3)$ mm/rev and depth of cut $t = (0.1; 0.2; 0.3; 0.4)$ mm. Each of the samples is correlated with one of the classes of surface quality: class 1 is for «clean» surface, class 2 is for intermediate surface and class 3 is for surface with traces of vibrations. According to the experimental data $Y$ vector of indexes of classes with size 1x36 is formed. This vector then transforms into matrix of target classes $T$ with size 3x36. The same method has been applied for defining chip morphology. Results of training, validating and testing of PNN for surface quality classification and chip morphology coincide with experimental data completely after retraining the network with problematic data that created an error in testing.

In LVQ network, the hidden layer has neurons with competitive activation function. Its task is to classify to input vectors subcluster. The output layer has neurons with linear activation functions for transformation of subclusters in the hidden layer as target classes of the output layer.

Learning of the network is produced through setting of the hidden layer weights for the most accurate subclustering to target classes $T_k$ of the input vector computed at $q$ iteration as follows:

$$ W_{I_k}(q) = W_{I_k}(q-1) + \mu (x(q) - W_{I_k}(q-1)), a_{2k} = T_k = 1; $$

$$ W_{I_k}(q) = W_{I_k}(q-1) + \mu (x(q) - W_{I_k}(q-1)), (a_{2k} = 1) \# (T_k = 0). $$

Both the competitive and linear layers of LVQ network have one neuron per class. Linear activation functions «line» transforms the vectors of the hidden layer into target classes of the output layer:

$$ \text{line}(n) = n. $$

Source data are the same as PNN case. Results of modeling are also successful.

Two-layer feedforward network also can be trained to classify the technological mode to surface classes or chip morphology. The hidden layer has 12 neurons with logistic sigmoidal activation function «sigmoid»:
\[
\text{logsig}(n) = \frac{1}{1 + e^{-n}}.
\]  
(10)

The number of neurons was obtained from preliminary studies. Output layer has three neurons corresponding with three target classes with softmax activation function («softmax»):

\[
a_j = \frac{e^{y_j}}{\sum_{k=1}^{N} e^{y_k}},
\]  
(11)

where \(j\) is the number of neuron in output layer.

The results of modeling in three cases are similar and successfully solve the problem with the network learning method described above.

While obtaining estimates of the surface roughness class and chip morphology type in modeling all three types of networks, errors were observed in test samples (no more than 4%). After exchanging the problematic data in the training sample the network testing was provided without errors. The making decision on the choice of processes mode in terms of roughness, taking into account the type of chips is necessary to ensure the cleanliness of the surface with high reliability. As shown by experiments in the formation of discontinuous chips for the surface «clean» class was determined in only one case, while the other two types of chips were found for «clean» class equally in 5 experiments each. In addition, if the ANNs indicate a possible formation of ribbon chips, it is necessary to provide for measures for chip breaking.

6. Results and conclusion

The results of successful classification of cutting modes according to the criterion of their correlation to the correct class of surface quality and chip morphology type obtained during hard turning of 105WCr6 steel are described in the article. Next types of networks were used: probabilistic neural network, Learning Vector Quantization network and two-layer feedforward network. All this networks can correctly define machined surface quality and chip morphology depending on cutting mode. Comparison of the simulation results of surface cleanliness and chip type is necessary to confirm that the required roughness will be obtained. If the type of chips for this cutting mode is defined as discontinuous or snarled, even with the resulting surface quality parameter-clean, because of the more intense vibrations the roughness will be increased. The result can be used in creating of automated system for technological processes design or in creating of expert system for rapid and reliable choice of cutting mode in designing hard turning operations.

References
[1] Gaitonde V N, Karnik S R, Figueira L et al 2011 Int J Adv Manuf Technol 52 101–14 doi 10.1007/s00170-010-2714-3
[2] Labidi A, Tebassi H, Belhadi S et al 2018 J Fail Anal and Preven 18 1017–33 https://doi.org/10.1007/s11668-018-0501-x
[3] Sharma V S, Dhiman S, Sehgal R et al 2008 J Intell Manuf 19 473–83 DOI 10.1007/s10845-008-0097-1
[4] Gupta A K, Guntuku S C, Desu R K et al 2015 Int J Adv Manuf Technol 77 331–9
[5] Madić M and Radovanović M 2013 J Braz. Soc. Mech. Sci. Eng. 35 111–21
[6] Aydın M, Karakuzu C, Uçar M et al 2013 Int J Adv Manuf Technol 67 957–67 DOI 10.1007/s00170-012-4540-2
[7] Rastorguev D A and Sevastyanov A A 2018 Science Vector of Togliatti State University 46 24–32 doi: 10.18323/2073-5073-2018-4-24-32
[8] Rastorguev D A, Zhotov A V and Bobrowskii A V 2017 ICIE 1443–51
[9] Haykin, S 2016 Neural Networks. A Comprehensive Foundation (Hamilton: McMaster University)