Optimal Selection of ANN Training and Architectural Parameters Using Taguchi Method: A Case Study

Miloš J. Madić
Research Assistant
University of Niš
Faculty of Mechanical Engineering

Miroslav R. Radovanović
Full Professor
University of Niš
Faculty of Mechanical Engineering

In recent years, there is increasing interest in using artificial neural networks (ANNs) for modeling and optimization of machining processes. The advantages that ANNs offer are numerous and are achievable only by developing an ANN model of high performance. However, determining suitable training and architectural parameters of an ANN still remains a difficult task. These parameters are typically determined in trial and error procedure, where a large number of ANN models are developed and compared to one another. This paper presents the application of Taguchi method for the optimization of ANN model trained by Levenberg-Marquardt algorithm. A case study of a modeling resultant cutting force in turning process is used to demonstrate implementation of the approach. The ANN training and architectural parameters were arranged in L_{18} orthogonal array and the predictive performance of the ANN model is evaluated using the proposed equation. Using the analysis of variance (ANOVA) and analysis of means (ANOM) optimal ANN parameter levels are identified. Taguchi optimized ANN model has been developed and has shown high prediction accuracy. Analyses and experiments have shown that the optimal ANN training and architectural parameters can be determined in a systematic way, thereby avoiding the lengthy trial and error procedure.

Keywords: *ANN, optimization, Taguchi method.*

1. INTRODUCTION

For more than 15 years, artificial neural networks (ANNs) have been widely applied in manufacturing industry for modeling and optimization of various processes. In the area of machining, recent papers [1-8] confirm the validity and effectiveness of using ANNs as promising and most powerful computer modeling techniques.

Benardos and Vosniakos [1] used ANN for the prediction of surface roughness in CNC face milling using the experimental data collected according to the principles of Taguchi design of experiments. The factors considered in the experiment were the depth of cut, the feed rate per tooth, the cutting speed, the engagement and wear of the cutting tool, the use of cutting fluid and the three components of the cutting force. Sharma et al. [2] developed ANN model for estimation of cutting forces and surface roughness in hard turning. Approaching angle, cutting speed, feed rate and depth of cut were selected as ANN inputs. In order to determine optimal ANN model number of hidden neurons (10, 20, 30, 40, 50) in single hidden layer was varied as well as the number of training iterations (100, 200, 300, 400, 500) in one-factor-at-a-time experimentation. After training, each ANN model is tested with the testing data, and optimal ANN architecture was found using the linear programming method by minimizing test error with testing data, minimizing training time and mean square error for training data. The optimal ANN model having 4 $[20]_1$ 4 architecture achieved overall 76.4 % accuracy. Ezugwu et al. [3] developed ANN models for modeling the correlation between cutting and process parameters in high-speed machining of Inconel 718 alloy. In order to determine the optimal ANN architecture, single and double hidden layer networks with 10 and 15 hidden neurons were considered. The networks were trained with Levenberg-Marquardt (LM) algorithm with Bayesian regularization and early stopping procedure. In one-factor-at-a-time experimentation, eight ANN models were developed and trained. ANN model with 4 $[10\text{-}10]_2$ 1 architecture trained with LM algorithm and Bayesian regularization was chosen as the optimum model. The ANN model yielded correlation coefficient between the model prediction and experimental values ranging from 0.6595 for cutting force to 0.9976 for nose wear prediction. Hans Ray et al. [4] showed some advantages of training ANNs using LM algorithm over the standard backpropagation (BP) algorithm in modeling of metal forming and metal cutting processes. They used double hidden layers ANNs with equal number of neurons in both hidden layers. The results obtained are found to correlate well with the finite element simulation data in cases of metal forming, and experimental data in case of metal cutting. Zain et al. [5] applied ANNs for developing the prediction model for surface roughness in the end milling. Three cutting parameters (cutting speed, feed rate and rake angle) were considered as ANN inputs. The determination of the number of layers and neurons in the hidden layers is done by the trial-and-error method considering some guidelines from literature. Eight ANN models were developed and trained. However, as in [3,4] the ANN model with two hidden layers had equal number

Received: February 2011, Accepted: April 2011
Correspondence to: Miloš Madić, Dipl.-Ing.
Faculty of Mechanical Engineering,
Aleksandra Medvedeva 14, 18000 Niš, Serbia
E-mail: madic@masfak.ni.ac.rs

of neurons in both hidden layers. With a total sample size of 24, separated into 16 samples for training and 8 samples for testing, it was shown that 3[1]$_1$ 1 ANN model is the best model which is capable of accurate predictions. Karayel [6] presented ANN approach for the prediction and control of surface roughness in a computer numerically controlled (CNC) lathe. Based on full factorial experimentation (4^3), considering depth of cut, cutting speed, and feed rate as cutting parameters, ANN models were developed and trained using the scaled conjugate gradient algorithm (SCGA) with adaptive learning rate. The number of hidden neurons in single hidden layer ANN models was determined by trial and error method. Tsao and Hocheng [7] applied multiple regression analysis (MRA) and radial basis function network (RBFN) for the prediction and evaluation of thrust force and surface roughness in drilling of composite material. Based on a L_{27} (3^{13}) Taguchi's orthogonal array (OA) three production parameters (feed rate, spindle speed and drill diameter) and their interactions were investigated. The data was then used for model development using MRA and ANN approach. The RBFN training and architectural parameters were determined by trial and error method. However, it was demonstrated that RBFN is much more accurate than MRA. Çaydaş and Hasçalik [8] developed ANN and MRA model for prediction of surface roughness in abrasive waterjet machining (AWJ) process. In the development of predictive models, machining parameters of traverse speed, waterjet pressure, standoff distance, abrasive grit size and abrasive flow rate were considered as model variables. Optimal 13 [22]$_1$ 1ANN architecture was determined by trial and error method. The mean error of ANN predictions was 3.0072 %.

The wide usage of ANNs is due to their ability to learn (through training process) complex nonlinear and multi input/output relationships between process parameters using the process data. The ANNs have many other advantageous characteristics, which include: generalization, adaptation, universal function approximation, parallel data processing, robustness, etc. Multilayer perceptron (MLP) trained with BP algorithm is the most used ANN in modeling and optimization of machining processes. Although BP algorithm has proved to be efficient, its convergence tends to be very slow, and there is a possibility to get trapped in some undesired local minimum. The LM algorithm is by far the fastest algorithm for moderate-sized (up to several hundred free parameters) MLPs [9], and offers some additional advantages in ANN training.

The quality of the developed ANN is highly dependable not only on ANN training algorithm and its parameters but also on many ANN architectural parameters. Above all, there is limited theoretical and practical background to assist in systematical selection of ANN parameters through entire ANN development and training process. Because of that, ANN parameters are usually set by previous experience in trial and error procedure which is very time consuming. In such a way the optimal settings of ANN parameters for achieving best ANN quality are not guaranteed.

The robust design methodology, proposed by Taguchi, is one of the appropriate methods for achieving this goal.

There is a wide range of applications of Taguchi method (TM) for the optimization of various processes in engineering. For instance, Dura and Isac [10] illustrated the application of TM for the quality improvement of the electrochemical cadmium plating in the drum in order to protect the representative parts of the hydraulic mining equipment. Thakur and Nandedkar [11] applied TM do determine the effect of process parameters (pressure, weld time and current) on tensile shear strength of resistance weld joint of austenitic stainless steel AISI 304. Optimum welding parameters determined by TM improved welding strength. Using the TM, Syrcos [12] analyzed and determined optimal settings of five casting parameters (piston velocity in first and second stage, metal temperature, filling time and hydraulic pressure) of the die casting method of AlSi9Cu13 aluminum alloy.

The application of TM for the design of experiments in the selection of optimal operating conditions in machining can be found in references [1,13-15].

The application of TM for ANN training data collection is presented in references [1,7,8].

However, there are limited number of papers related to the application of TM to selection of ANN training and architectural parameters.

Benardos and Vosniakos [1] used L_{32} Taguchi's OA in order to select the most influential combination of factors that would be used as ANN inputs in order to develop accurate surface roughness model.

Wang et al. [16] applied TM to identify "best" ANN structure for process cost modeling. Six design factors were considered (summation functions, noise function, transfer function, output function, error function and learning rules) and arranged in L_{18} ($2^1 \times 3^7$) OA. The authors concluded that ANN efficiency depends on: the selection of the data for training and testing the ANN, the order in which data are presented during training process and number of hidden layers, number of neurons in each layer. Sukthomya and Tannock [17] used Taguchi's approach for the parameter setting and optimization of a multilayer ANN trained with a BP algorithm. The authors showed that inclusion of control factors like proportion of testing data and the use of a training hint can have a significant effect on ANN performance. Khaw et al. [18] applied TM to determine the number of hidden layers, the number of neurons in a hidden layer, and the size of the training set to increase the accuracy and convergence speed of ANN trained with BP algorithm. It turned out that the most important factors were input representation scheme and training sample size.

This paper describes the application of TM for the ANN parameter level optimization of a MLP network, trained with LM algorithm. In order to develop ANN model of high performance, the parameters related to ANN training as well as the architecture parameters are considered. A case study using real experimental data is presented to illustrate the technique and its outcomes.

2. ARTIFICIAL NEURAL NETWORK (ANN) DESIGN ISSUES

2.1 ANN architectural parameters

Besides many advantages that ANNs offer there are some drawbacks and limitations related to ANN design process. In order to develop an ANN which generalizes well and is robust, one has to take into consideration a number of issues, particularly related to ANN architecture and ANN training parameters. The ANN design parameters that largely affect the ANN performance are illustrated in Figure 1.

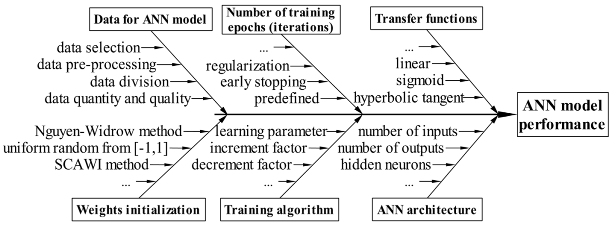

Figure 1. Cause-and-effects diagram for the ANN model performance

The detailed explanation and discussion about ANN and its parameters is beyond the scope of the paper. Good introduction about ANNs and applications can be found in [19-23].

Choosing the ANN architecture followed by selection of training algorithm and related parameters is rather a matter of the designer past experience since there are no practical rules which could be generally applied. This is usually a very time consuming trial and error procedure where a number of ANNs are designed and compared to one another. Above all, the design of optimal ANN is not guaranteed. It is unrealistic to analyze all combination of ANN parameters and parameter's levels effects on the ANN performance. To deal economically with the many possible combinations one can apply the TM. This paper describes the effective application of TM for the design of neural networks with high prediction accuracy considering the most important ANN training and architectural parameters.

2.2 ANN training parameters – Levenberg-Marquardt algorithm

Levenberg-Marquardt (LM) algorithm is an iterative technique that locates a local minimum of a multivariate function that is expressed as the sum of squares of several non-linear, real-valued functions [24]. The algorithm changes current weights of the network iteratively such that objective function, $F(w)$, is minimized as:

$$F(w)=\sum_{i=1}^{P}\sum_{j=1}^{M}\left(d_{ij}-o_{ij}\right)^{2}, \tag{1}$$

where: $w = [w_1\ w_2\ ...\ w_N]^T$ is a vector of all weights and N is the number of weights, P is the number of patterns (observations), M is the number of output neurons, d_{ij} and o_{ij} are the desired value ("target value") and the actual value ("predicted value") of the i-th output neuron and the j-th pattern. Equation (1) can be also written as:

$$F(w)=EE^{T}, \tag{2}$$

where: $E = [e_{11}...\ e_{m1},\ e_{12}\ ...\ e_{m2},\ ...\ e_{1P}\ ...\ e_{MP}]^T$, $e_{mp} = d_{mp} - o_{mp}$, $m = 1\ ...\ M$, $p = 1\ ...\ P$, E is the cumulative error vector. The increments of weights are done according to:

$$\Delta w=\left(J^{T}J+\mu I\right)^{-1}J^{T}E, \tag{3}$$

where: J is Jacobian matrix of derivatives of each error to each weight, μ is learning parameter (small scalar which controls the learning process), and I is identity matrix. In this way, the weight updates is based on the following equation:

$$w_{t+1}=w_{t}+\left[J_{t}^{T}J_{t}+\mu I\right]^{-1}J_{t}^{T}E_{t}. \tag{4}$$

The learning parameter μ changes during the training. Starting with an initial value of μ, the μ is decreased after each successful step ($F(w)$ decreases) by decrements of $\Delta\mu^{-}$. Elsewhere, if $F(w)$ increases, it is increased by $\Delta\mu^{+}$. When μ is small, the method is called second order Newton's, while when set to a large number, it is called gradient descent with small step size. Training continues until the error goal is met, μ reaches a maximum value, or the maximum number of epochs is completed. In practice, LM algorithm is faster and finds better optima for a variety of problems than do the other usual methods. This high performance algorithm can converge from ten to one hundred times faster than the conventional algorithms [9].

3. TAGUCHI METHOD

The Taguchi technique is a methodology for finding the optimum setting of the control factors to make the product or process insensitive to the noise factors [25,26]. Taguchi based optimization technique has produced a unique and powerful optimization discipline that differs from traditional practices. Taguchi's techniques have been used widely in engineering design [27], and can be applied to many aspects such as optimization, experimental design, sensitivity analysis, parameter estimation, model prediction, etc. The distinct idea of Taguchi's robust design that differs from the conventional experimental design is that of designing for the simultaneous modeling of both mean and variability [25]. However, Taguchi methodology is based on the concept of fractional factorial design [28]. By using OAs and fractional factorial instead of full factorial, Taguchi's approach allows for an easy set-up of experiments with a very large number of factors varied on few levels [29]. The two major goals of parameter design are to minimize the process or product variation and to design robust and flexible processes or products that are adaptable to environmental conditions.

Taguchi method uses a special design of OAs to study the entire parameter space with a small number of experiments. An OA is a small fraction of full factorial design and assures a balanced comparison of levels of any factor or interaction of factors. The columns of an OA represent the experimental parameters to be optimized and the rows represent the individual trials (combinations of levels). The array is called orthogonal because for every pair of parameters, all combinations of parameter levels occur an equal number of times. The mean and the variance of the response at each setting of parameters in OA are then combined into a single performance measure known as the signal-to-noise (S/N) ratio [25].

The S/N ratio is a quality indicator by which the experimenters can evaluate the effect of changing a particular experimental parameter on the performance of the process or product. Depending on the criterion for the quality characteristic to be optimized, different S/N ratios can be chosen: smaller-the-better, larger-the-better, and nominal-the-better.

A full explanation of the method can be found in references [25,26].

4. APPLICATION OF TAGUCHI METHOD TO ANN DESIGN

In order to illustrate the use of the Taguchi method for neural network design, a case study for the design of a feed-forward ANN trained with LM algorithm for prediction of resultant cutting force in turning [30] is used.

The ANN consists of 4 input neurons which correspond to four cutting parameters (depth of cut – a_p, cutting speed – v, feed rate – f and cutting edge angle – κ), and one output neuron that corresponds to the resultant cutting force – F_R (Fig. 2).

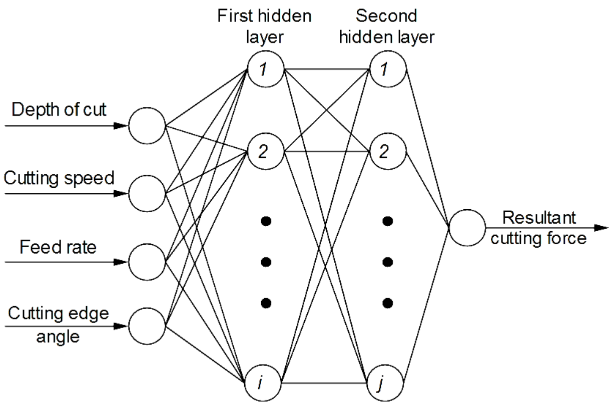

Figure 2. General architecture of the ANN prediction model

The experimental data used for ANN models development are given in Table 1. To train the ANN models, 30 data from Table 1 were used and the rest are used for model testing. The selection of data for training and testing was made by random method.

Table 1. Experimental data

a_p [mm]	v [m/min]	f [mm/rev]	F_R [N]		
			κ [°]		
			75	85	95
1.5	143	0.499	1648*	1654	1874
0.75	143	0.499	944	979	952
1.5	143	0.124	620	630	668
0.75	143	0.124	365	387*	383
1.5	94	0.499	1844	1759	1904*
0.75	94	0.499	1024	1073	986
1.5	94	0.124	639	679	721
0.75	94	0.124	375	409	418
1	116	0.249	773	774	815
1	116	0.249	763*	784	813
1	116	0.249	773	788*	821*
1	116	0.249	785	766	829

* denotes data for ANN testing

The main objective of the proposed Taguchi based optimization is to develop accurate and robust ANN model. In other words, the goal is to select ANN training and architectural parameters, so that the ANN model yields best performance.

The performance of the ANN is evaluated using a proposed performance index (PI):

$$PI = R - \left(RMSE_{\text{tr}} + RMSE_{\text{ts}} \right), \qquad (5)$$

where: R is the correlation coefficient obtained for ANN predictions and experimental values using whole data, and $RMSE$ is root mean squared error on training and testing data. Since the ANN accuracy belongs to a larger-the-better type problem its corresponding objective function to be maximized is represented by the following equation:

$$\eta = -10 \log_{10} \left(\frac{1}{n} \sum_{i=1}^{n} \frac{1}{y_i^2} \right), \qquad (6)$$

where: η is the S/N ratio (in dB) for the ANN accuracy, n is the number of measurements, and y_i is i-th observed response value.

The design control parameters that influence the neural performance are identified and divided into two groups: ANN architectural parameters (number of neurons in first and second hidden layer, transfer function in hidden and output layer) and parameters of LM algorithm (learning parameter, increment and decrement factor).

Random initialization of weights is considered as noise factor. The weights are initialized within [–0.1, 0.1] interval from uniform distribution, within [–1, 1] interval from uniform distribution and by Nguyen-Widrow (N-W) method, respectively.

The ANN design parameters and the corresponding levels are shown in Table 2. This design problem has eight main parameters, where one has 2 levels and other seven 3 levels. If all of the possible combinations of the eight parameters were to be considered, then $2^1 \cdot 3^7$ = 4374 different ANN models would have to be created.

Table 2. ANN training and architectural parameter and levels

Parameter	Parameter description	Level 1	Level 2	Level 3
A	Transfer function in output layer	tansig	purelin	–
B	Hidden neurons in first layer	2	4	8
C	Hidden neurons in second layer	0	2	4
D	Learning parameter, μ	0.001	0.01	0.1
E	increment factor, $\Delta\mu^+$	5	10	15
F	decrement factor, $\Delta\mu^-$	0.05	0.1	0.2
G	Transfer function in hidden layers	tansig[1]	logsig[1]	purelin[1]
H	Initial weights	[–0.1, 0.1]	[–1, 1]	N-W

[1] MATLAB function for the corresponding transfer function

This is unrealistic, so by using Taguchi's OA this number is significantly reduced. As the total degree of freedom (DoF) for the eight parameters is $1 + 7 \cdot 2 = 15$, a mixed OA $L_{18}(2^1 \times 3^7)$ [25] was used for experimentation as it has 17 DoF which is more than DoF of selected ANN design parameters.

For each experiment corresponding to each row of the L_{18} OA, two replications are used. Thus a total of 36 experiments are conducted in order to assess the sensitivity of the ANN performance. The levels in the L_{18} OA and the *S*/*N* ratios (η) of each experiment are given in Table 3.

Table 3. L_{18} OA for the ANN parameter optimization

Trial No	A	B	C	D	E	F	G	H	*PI* 1	*PI* 2	η
1	1	1	1	1	1	1	1	1	0.792	0.962	–1.263
2	1	1	2	2	2	2	2	2	0.932	0.925	–0.645
3	1	1	3	3	3	3	3	3	0.734	0.733	–2.692
4	1	2	1	1	2	2	3	3	0.734	0.734	–2.686
5	1	2	2	2	3	3	1	1	0.963	0.958	–0.350
6	1	2	3	3	1	1	2	2	0.977	0.93	–0.422
7	1	3	1	2	1	3	2	3	0.798	0.93	–1.346
8	1	3	2	3	2	1	3	1	0.742	0.735	–2.633
9	1	3	3	1	3	2	1	2	0.969	0.868	–0.778
10	2	1	1	3	3	2	2	1	0.955	0.96	–0.377
11	2	1	2	1	1	3	3	2	0.691	0.719	–3.041
12	2	1	3	2	2	1	1	3	0.899	0.952	–0.683
13	2	2	1	2	3	1	3	2	0.711	0.709	–2.975
14	2	2	2	3	1	2	1	3	0.88	0.905	–0.990
15	2	2	3	1	2	3	2	1	0.967	0.95	–0.369
16	2	3	1	3	2	3	1	2	0.853	0.713	–2.229
17	2	3	2	1	3	1	2	3	0.889	0.953	–0.731
18	2	3	3	2	1	2	3	1	0.711	0.709	–2.975

5. ANALYSIS OF RESULTS

5.1 Analysis of means

Analysis of means (ANOM) is the process of estimating the factor effects. First, the overall mean value of η for the experimental region defined by the parameters levels in Table 2 is given by:

$$m = \frac{1}{18}\sum_{i=1}^{18}\eta_i \ , \tag{7}$$

where the subscript *i* represents the *i*-th experiment in the OA. The average η of the experiments for factor A at level 1 can be calculated according to the level assignment in OA [25] as:

$$m_{A_1} = \frac{1}{9}\left(\eta_1 + \eta_2 + \eta_3 + \eta_4 + \eta_5 + \eta_6 + \eta_7 + \eta_8 + \eta_9\right). \tag{8}$$

The other parameters at different levels can be calculated in a similar way. Table 4 lists the mean effects of each design parameter in which the optimal level with the highest signal-to-noise ratio is bolded. The optimum level for a factor is the level that gives the highest value of η in the experimental region [25].

Table 4. Analysis of means (ANOM)

ANN parameters	Level 1	Level 2	Level 3
A	**–1.4238**	–1.5967	
B	–1.4502	**–1.2987**	–1.7819
C	–1.8126	–1.3984	**–1.3198**
D	**–1.4779**	–1.4956	–1.5573
E	–1.6728	–1.5409	**–1.3171**
F	–1.4510	**–1.4085**	–1.6713
G	–1.0489	**–0.6482**	–2.8337
H	**–1.3279**	–1.6815	–1.5213

These averages are also shown graphically in Figure 3 where optimal levels are marked with circles.

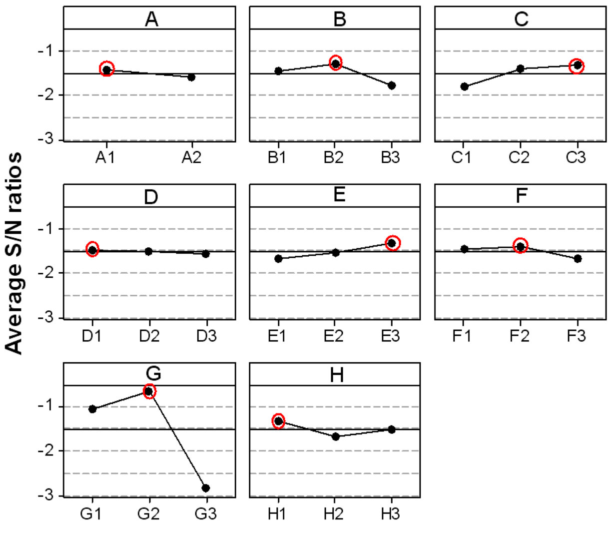

Figure 3. ANN parameters influence on performance variation

From the Table 4 and Figure 3, one can observe that the optimal ANN parameter levels are $A_1B_2C_3D_1E_3F_2G_2H_1$. In other words, ANN having 4 hidden neurons in first and second hidden layer, using logsig transfer function in hidden layers and tansig transfer function in output layer, and trained with LM algorithm using $\mu = 0.001$ as initial learning parameter, $\mu^- = 0.1$ as decrement factor and $\mu^+ = 15$ as increment factor, where training was started with weights initialized randomly from the interval [–0.1 and 0.1], is the optimal ANN model.

5.2 Analysis of variance

Analysis of variance (ANOVA) was performed using the *S*/*N* ratios as the response (Table 5). The purpose of ANOVA is to determine the relative magnitude of the effect of each factor on the objective functions, η, and to estimate the error variance.

It can be seen from Table 5 that changing the design parameters B, C and G between the three chosen levels contributes to 92.6 % of the total variation in the ANN performance.

Table 5. Analysis of variance (ANOVA) for *S/N* ratios

Source	DOF	SS	MS	F	Contribution
A	1	0.1347	0.1347	1.013458071	0.6997348
B	2	0.7329	0.3665	2.758051705	**3.8085539**
C	2	0.8414	0.4207	3.166097914	**4.372019**
D	2	0.0208	0.0104	0.078339974	0.1081785
E	2	0.3879	0.1939	1.459603033	2.0155448
F	2	0.2387	0.1194	0.898326018	1.2404855
G	2	16.2463	8.1232	61.1343067	**84.419483**
H	2	0.3763	0.1881	1.41584107	1.9551145
Error	2	0.2657	0.1329		1.3808856
Total	17	19.2448			

5.3 Confirmation experiment

Confirmation testing is necessary and important step in the TM. In addition to the variance analysis, the optimal ANN design parameters have to be verified to see if the error caused by the interaction among the design parameters is within an acceptable tolerance. The expected response for the best design can be calculated and confirmed through the confirmation test.

Taguchi prediction of η under optimum conditions can be estimated by:

$$\eta_{\text{est}} = m + \left(m_{A_1} - m\right) + \left(m_{B_2} - m\right) + \\ + \left(m_{C_3} - m\right) + \left(m_{D_1} - m\right) + \left(m_{E_3} - m\right) + \\ + \left(m_{F_2} - m\right) + \left(m_{G_2} - m\right) + \left(m_{H_1} - m\right). \qquad (9)$$

Hence, the predicted *PI* under optimum conditions can be estimated by:

$$PI_{\text{est}} = 10^{-\frac{\eta_{\text{est}}}{20}}. \qquad (10)$$

There is no need to run the confirmation test if the best design is already included in the OA. However, the best design identified in this experiment is not included in the OA, and therefore a confirmation test is conducted. Using the $A_1B_2C_3D_1E_3F_2G_2H_1$ combination of ANN design parameter levels, the optimized ANN model is developed. With two repetitions, ANN model is trained and tested. It yielded the *PI* of 0.954 and 0.956, or combined in *S/N* ratio $\eta^* = 0.399$.

In order to judge the closeness of the η_{est} and η^*, one need to determine the variance of the prediction error, σ^2_{pred}. The variance of the prediction error has two components. The first is the error in prediction of η_{est} caused by the errors in the estimates of m, m_{A1}, m_{B2}, m_{C3}, m_{D1}, m_{E3}, m_{F2}, m_{G2} and m_{H1}. The second component represents the repetition error of an experiment. Thus, the variance of the prediction error can be calculated by:

$$\sigma^2_{\text{pred}} = \left(\frac{1}{n_0}\right)\sigma_e^2 + \left(\frac{1}{n_r}\right)\sigma_e^2, \qquad (11)$$

where equivalent sample size n_0 is calculated by:

$$\left(\frac{1}{n_0}\right) = \frac{1}{n} + \left(\frac{1}{n_{A_1}} - \frac{1}{n}\right) + \left(\frac{1}{n_{B_2}} - \frac{1}{n}\right) + \\ + \left(\frac{1}{n_{C_3}} - \frac{1}{n}\right) + \left(\frac{1}{n_{D_1}} - \frac{1}{n}\right) + \left(\frac{1}{n_{E_3}} - \frac{1}{n}\right) + \\ + \left(\frac{1}{n_{F_2}} - \frac{1}{n}\right) + \left(\frac{1}{n_{G_2}} - \frac{1}{n}\right) + \left(\frac{1}{n_{H_1}} - \frac{1}{n}\right), \qquad (12)$$

where: n is the number of rows in the OA, n_r is the number of repetitions of verification experiment, and n_{A1} is the number of times the level A_1 was repeated in OA. n_{B2}, n_{C3}, n_{D1}, n_{E3}, n_{F2}, n_{G2}, and n_{H1} are calculated in the same way. In statistical analysis, the two-sigma confidence is defined for the prediction error. Using the (11) and (12), the two standard deviation confidence limits for the prediction error, $2 \cdot \sigma_{\text{pred}}$, are calculated and are ±0.7876 dB. Since the $|\eta_{\text{est}} - \eta^*| = 0.05 < 2 \cdot \sigma_{\text{pred}}$, the optimal ANN model can be validated. In other words, the interaction between the ANN design parameters is within acceptable limits of two-sigma confidence.

6. PREDICTION OF RESULTANT CUTTING FORCE

Once the Taguchi optimized ANN model is developed, it can be used for the prediction of resultant cutting force. There are a number of statistics for measuring the accuracy of prediction models and each has advantages and limitations. However, (Pearson's) correlation coefficient is the most widely used statistic. The correlation coefficient is a statistical measure of the strength of correlation between actual versus predicted values. For example, the value of + 1 indicates perfect correlation. In that case, all points should lie on the line passing through the origin and inclined at 45°. The high performance of optimized ANN model is confirmed by very high correlation coefficient between experimental and predicted resultant cutting force values as shown in Figure 4.

7. CONCLUSION

This article has described the application of Taguchi method for the selection of ANN parameters. The eight ANN training and architectural parameters have been identified and were arranged in the L_{18} OA. Analyses show that the transfer function in hidden layer (G) is most influential on ANN prediction performance (84.42 %), which was measured by proposed equation. This can be explained considering that ANN can handle non-linear relationships between input and output variables using sigmoid transfer functions in a hidden layer. The remaining design parameters have minor effects. In other words, the selected design factors and their levels had little effect of ANN prediction accuracy.

It is found that the best ANN model architecture had four hidden neurons in both hidden layers. This confirms the findings from [3,4]. Analysis shows that adding more neurons in a first hidden layer has negative effect of ANN performances. This finding further supports the conclusion by Ćirović and Aleksendrić [31] that too many neurons in the first hidden layer are not desirable when training ANNs with LM algorithm.

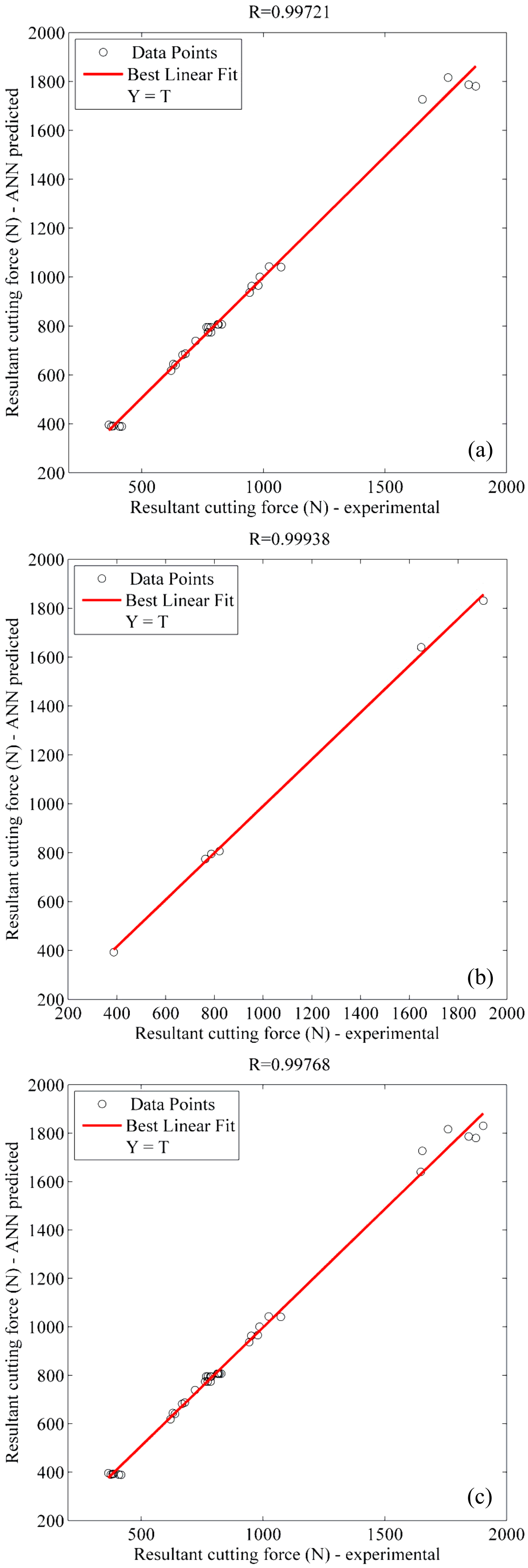

Figure 4. The performance of Taguchi optimized ANN model for prediction of cutting force in turning: (a) for training data set, (b) for testing data set, and (c) for whole data set

Additionally, it was shown that LM algorithm is capable of fast ANN training for finding good minima starting from different initial weights ranges.

In the authors' opinion, the optimal selection of ANN training and architectural parameters is largely problem dependent. However, it was shown that Taguchi method can be successfully applied in ANN design and training in order to develop ANN model of high performance with a relatively small and time-saving experiment.

The methodology presented in this paper might be utilized in any ANN, for each ANN application.

ACKNOWLEDGMENT

This work was carried out within the project TR 35034 financially supported by Ministry of Science and Technological Development of the Republic of Serbia.

REFERENCES

[1] Benardos, P.G. and Vosniakos, G.C.: Prediciton of surface roughness in CNC face milling using neural networks and Taguchi's design of experiments, Robotics and Computer-Integrated Manufacturing, Vol. 18, No. 5-6, pp. 343-354, 2002.

[2] Sharma, V.S., Dhiman, S., Sehgal, R. and Sharma, S.K.: Estimation of cutting forces and surface roughness for hard turning using neural networks, Journal of Intelligent Manufacturing, Vol. 19, No. 4, pp. 473-483, 2008.

[3] Ezugwu, E.O., Fadare, D.A., Bonney, J., Da Silva, R.B. and Sales, W.F.: Modelling the correlation between cutting and process parameters in high-speed machining of Inconel 718 alloy using an artificial neural network, International Journal of Machine Tools and Manufacture, Vol. 45, No. 12-13, pp. 1375-1385, 2005.

[4] Hans Raj, K., Sharma, R.S., Srivastava, S. and Patvardhan, C.: Modeling of manufacturing processes with ANNs for intelligent manufacturing, International Journal of Machine Tools and Manufacture, Vol. 40, No. 6, pp. 851-868, 2000.

[5] Zain, A.M., Haron, H. and Sharif, S.: Prediction of surface roughness in the end milling machining using artificial neural network, Expert Systems with Applications, Vol. 37, No. 2, pp. 1755-1768, 2010.

[6] Karayel, D.: Prediction and control of surface roughness in CNC lathe using artificial neural network, Journal of Materials Processing Technology, Vol. 209, No. 7, pp. 3125-3137, 2009.

[7] Tsao, C.C. and Hocheng, H.: Evaluation of thrust force and surface roughness in drilling composite material using Taguchi analysis and neural network, Journal of Materials Processing Technology, Vol. 203, No. 1-3, pp. 342-348, 2008.

[8] Çaydaş, U. and Hasçalik, A.: A study on surface roughness in abrasive waterjet machining process using artificial neural networks and regression analysis method, Journal of Materials Processing Technology, Vol. 202, No. 1-3, pp. 574-582, 2008.

[9] Beale, M.H., Hagan, M.T. and Demuth, H.B.: *Neural Network Toolbox User's Guide*, The Mathworks Inc, Massachusetts, 2004.

[10] Dura, C. and Isac, C.: Using Taguchi methods to improve the production process quality: A case study, Total Quality Management & Business Excellence, Vol. 20, No. 11, pp. 1189-1212, 2009.

[11] Thakur, A.G. and Nandedkar, V.M.: Application of Taguchi method to determine resistance spot welding conditions of austenitic stainless steel AISI 304, Journal of Scientific & Industrial Research, Vol. 69, No. 9, pp. 680-683, 2010.

[12] Syrcos, G.P.: Die casting process optimization using Taguchi methods, Journal of Materials Processing Technology, Vol. 135, No. 1, pp. 68-74, 2003.

[13] Bagci, E. and Aykut, Ş.: A study of Taguchi optimization method for identifying optimum surface roughness in CNC face milling of cobalt-based alloy (stellite 6), International Journal of Advanced Manufacturing Technology, Vol. 29, No. 9-10, pp. 940-947, 2006.

[14] Ghani, J.A., Choudhury, I.A. and Hassan, H.H.: Application of Taguchi method in the optimization of end milling parameters, Journal of Materials Processing Technology, Vol. 145, No. 1, pp. 84-92, 2004.

[15] Deng, C.-S. and Chin, J.-H.: Hole roundness in deep-hole drilling as analysed by Taguchi methods, International Journal of Advanced Manufacturing Technology, Vol. 25, No. 5-6, pp. 420-426, 2005.

[16] Wang, Q., Stockton, D.J. and Baguley, P.: Process cost modelling using neural networks, International Journal of Production Research, Vol. 38, No. 16, pp. 3811-3821, 2000.

[17] Sukthomya, W. and Tannock, J.: The optimisation of neural network parameters using Taguchi's design of experiments approach: an application in manufacturing process modelling, Neural Computing & Applications, Vol. 14, No. 4, pp. 337-344, 2005.

[18] Khaw, J.F.C., Lim, B.S. and Lim, L.E.N.: Optimal design of neural networks using the Taguchi method, Neurocomputing, Vol. 7, No. 3, pp. 225-245, 1995.

[19] Miljković, Z.: *Systems of Artificial Neural Networks in Production Technologies*, Faculty of Mechanical Engineering, University of Belgrade, Belgrade, 2003, (in Serbian).

[20] Miljković, Z. and Aleksendrić, D.: *Artificial Neural Networks: Solved Examples with Short Theory Background*, Faculty of Mechanical Engineering, University of Belgrade, Belgrade, 2009, (in Serbian).

[21] Haykin, S.: *Neural Networks: Comprehensive Foundation*, Prentice Hall, New York, 1998.

[22] Bose, N.K. and Liang, P.: *Neural Network Fundamentals With Graphs, Algorithms, and Applications*, Mc Graw-Hill, New York, 1996.

[23] Dreyfus, G.: *Neural Networks*, Springer Verlag, Berlin, 2005.

[24] Desta Zahlay, F. and Rama Rao K.S.: Autoreclosure in extra high voltage lines using Taguchi's method and optimized neural networks, in: *Proceedings of the International Conference on Computer Engineering and Technology 2009*, 22-24.01.2009, Singapore, pp. 151-155.

[25] Phadke, M.S.: *Quality Engineering Using Robust Design*, Prentice Hall, Upper Saddle River, 1989.

[26] Ross, P.J.: *Taguchi Techniques for Quality Engineering*, McGraw-Hill, New York, 1996.

[27] Taguchi, G.: *Introduction to Quality Engineering*, Asian Productivity Organization, Tokyo, 1986.

[28] Montgomery, D.C.: *Design and Analysis of Experiments*, John Wiley & Sons, New York, 1997.

[29] Sibalija, T.V., Majstorovic, V.D. and Miljkovic, Z.D.: An intelligent approach to robust multi-response process design, International Journal of Production Research, Article in Press.

[30] Radovanović, M., Madić, M. and Janković, P.: Artificial neural network modeling of cutting force components by turning, in: *Proceedings of the International Scientific Conference UNITECH '08*, 21-22.11.2008, Gabrovo, Bulgaria, pp. II-486-II-490.

[31] Ćirović, V. and Aleksendrić, D.: Development of neural network model of disc brake operation, FME Transactions, Vol. 38, No. 1, pp. 29-38, 2010.

ОПТИМАЛАН ИЗБОР ПАРАМЕТАРА АРХИТЕКТУРЕ И ТРЕНИРАЊА ВЕШТАЧКЕ НЕУРОНСКЕ МРЕЖЕ ПРИМЕНОМ ТАГУЧИ МЕТОДЕ: СТУДИЈА СЛУЧАЈА

Милош Ј. Мадић, Мирослав Р. Радовановић

Последњих година постоји све веће интересовање за примену вештачких неуронских мрежа (ВНМ) за моделирање и оптимизацију процеса обраде. Предности које нуде ВНМ су бројне и оне се постижу креирањем модела ВНМ високих перформанси. Међутим, одређивање параметара архитектуре и тренирања ВНМ и даље је тежак задатак. Обично се ови параметри одређују поступком тзв. „пробе и грешке“ где се велики број модела ВНМ креира и међусобно упоређује. У овом раду је приказана примена Тагучи методе за оптимизацију параметара ВНМ која се тренира Левенберг-Маркеовим алгоритмом. Примена овог приступа илустрована је на студија случаја за моделовање резултујућег отпора код стругања. Параметри ВНМ су распоређени у ортогонални низ L_{18}, а предикционе перформансе ВНМ модела су рачунате према предложеној једначини. Применом анализе варијансе и средњих вредности оптимални нивои параметара ВНМ су идентификовани. Затим је креиран Тагучи оптимизован модел ВНМ који је показао високу тачност предикције. Анализе и експерименти показују да се избор оптималних параметара ВНМ може одредити на систематичан начин чиме се избегава дуготрајан поступак „пробе и грешке“.