Advances in Materials Research, *Vol. 6, No. 2 (2017) 169-184*
DOI: https://doi.org/10.12989/amr.2017.6.2.169

Modeling of surface roughness in electro-discharge machining using artificial neural networks

Liborio Cavaleri[1], George E. Chatzarakis[2], Fabio Di Trapani[1], Maria G. Douvika[3], Konstantinos Roinos[3], Nikolaos M. Vaxevanidis[4] and Panagiotis G. Asteris*[3]

[1]Department of Civil, Environmental, Aerospace and Materials Engineering (DICAM), University of Palermo, Palermo, Italy
[2]Department of Electrical and Electronic Engineering Educators, School of Pedagogical and Technological Education, Athens, Greece
[3]Computational Mechanics Laboratory, School of Pedagogical and Technological Education, Athens, Greece
[4]Laboratory of Manufacturing Processes & Machine Tools, School of Pedagogical and Technological Education, Athens, Greece

(Received November 21, 2016, Revised July 7, 2017, Accepted July 11, 2017)

Abstract. Electro-Discharge machining (EDM) is a thermal process comprising a complex metal removal mechanism. This method works by forming of a plasma channel between the tool and the workpiece electrodes leading to the melting and evaporation of the material to be removed. EDM is considered especially suitable for machining complex contours with high accuracy, as well as for materials that are not amenable to conventional removal methods. However, several phenomena can arise and adversely affect the surface integrity of EDMed workpieces. These have to be taken into account and studied in order to optimize the process. Recently, artificial neural networks (ANN) have emerged as a novel modeling technique that can provide reliable results and readily, be integrated into several technological areas. In this paper, we use an ANN, namely, the multi-layer perceptron and the back propagation network (BPNN) to predict the mean surface roughness of electro-discharge machined surfaces. The comparison of the derived results with experimental findings demonstrates the promising potential of using back propagation neural networks (BPNNs) for getting a reliable and robust approximation of the Surface Roughness of Electro-discharge Machined Components.

Keywords: artificial neural networks (ANNs); back propagation neural networks (BPNNs); mean surface roughness; electro-discharge machining (EDM); normalization

1. Introduction

Electro-Discharge Machining is the most widely used and successful technique, among the various non-conventional machining methods, for the high precision manufacturing of a plethora of conductive materials regardless of their mechanical properties. It has been to be a very efficient and effective method for producing complex geometries on difficult-to-work materials; however,

*Corresponding author, Professor, E-mail: panagiotisasteris@gmail.com

http://www.techno-press.org/?journal=amr&subpage=7 ISSN: 2234-0912 (Print), 2234-179X (Online)

there are several problems pertaining to the resulting surface roughness and texture which in turn affects the product quality and limits the possible applications. Since these problems are due to the random nature of surface formation, there is a lack of formal methods for analytically, estimating the resulting surface roughness. Instead, we rely and employ empirical methods,usually, based on multi-regression analysis, for predicting the roughness of EDM worked pieces (Petropoulos *et al.* 2004, Petropoulos *et al.* 2009, Vaxevanidis *et al.* 2013, Al-Ghamdi and Aspinwall 2014, Al-Ghamdi and Taylan 2015, Das *et al.* 2014, Kumar *et al.* 2014, Moghaddam and Kolahan 2015, Pattnaik *et al.* 2014, Porwal *et al.* 2014, Tang and Guo 2014, Pradhan and Das 2015, Pramanick *et al.* 2014, Rahman Khan *et al.* 2014, Sarkheyli *et al.* 2014, Wang *et al.* 2014 and Huang *et al.* 2015). Detailed and in-depth state-of-the-art reports can be found in Shrivastava and Dubey (2014).

The surface roughness describes the geometry of the surface to be machined and it is interrelated with surface texture and surface integrity. The surface roughness formation mechanism is a very complex procedure that mainly, depends on the machining process. (Vaxevanidis *et al.* 2014) Hence, it is very difficult to determine the surface roughness through analytical equations.

Artificial Neural Networks (ANNs) have emerged as a novel approach that has been applied in several areas of technology, especially, for problems where the input and output values cannot be directly connected by simple equations. One such area, is the manufacturing process, where this technique has been used, particularly, to EDM (Dini 1997). So far the relevant literature, mainly, includes mainly, publications, involving the application of ANNs, for determining the removal rate of the process, the optimization of its parameters as well as its on-line monitoring (for a review see Markopoulos *et al.* 2008). Other artificial intelligence methods such as fuzzy logic and genetic algorithms have also been used for modeling the EDM process (Wang *et al.* 2003).

In the present paper, we consider the application of ANNs for predicting of the surface roughness of electrical discharge machined surfaces is investigated. We train these models, using data from an extensive experimental research work in the surface integrity of EDMed steels. This paper follows up on a statistical multi-parameter surface roughness analysis already published (Petropoulos *et al.* 2004, Petropoulos *et al.* 2006). Three tool steels, namely AISI D2, P20 modified and premium H13 were tested. Based on a design-of-experiments methodology, we varied the pulse current Ie and the pulse-on time Ip, from roughing to near finishing, resulting to different pulse energies. The ANN models presented in this paper take use the workpiece material, the pulse current and the pulse-on time as input parameters in order to predict the center-line average Ra surface roughness. The suggested neural networks were trained with experimental data from the previous mentioned series of experiments. The proposed neural networks have been proven to be very successful, exhibiting very reliable predictions, and thus constituting a valid alternative to lengthy and costly experiments.

2. Architecture of artificial neural networks

This section summarizes the mathematical and computational aspects of artificial neural networks (ANNs). A special emphasis is given on a heuristic algorithm which is proposed for the development of a reliable and robust ANN that can predict the mean surface roughness of electro-discharge machined surfaces. ANNs are information processing models configured for a specific application through a training process. A trained ANN has learned to rapidly map a given input into the desired output quantities (similar to curve fitting procedures) and thereby can be used as meta-models enhancing the computational efficiency of a numerical analysis process. This major advantage of a trained ANN over conventional numerical analysis procedures like regression analysis, under the condition that the training and validation data cover the entire range of input parameters values, is that the results can be produced with much less computational effort (Hornik *et al.* 1989, Adeli 2991, Plevris and Asteris 2014 and 2015, Giovanis and Papadopoulos 2015, Mansouri and Kisi 2015, Asteris and Plevris 2013, 2016, Asteris *et al.* 2016, Zhang *et al.* 2016, Xu and Gao 2016, Mansouri *et al.* 2016, Asteris and Kolovos 2017, Asteris *et al.* 2017).

2.1 Back-propagation neural networks

In the present study, we use a Back-Propagation Neural Network (BPNN). This type of NN, works, as follows. During a training phase, the output of the network is compared with the correct answer to compute the error, based on a predefined error function. Then the error is fed back, from the output layer of the network, all the way down to the input layer, through the various intermediate-hidden layers of the network. Based on the error-the discrepancy between the computed and the desired output value, the algorithm adjusts the adaptive weights-the plastic contacts of the connections to the nodes, in each layer of the network. The algorithm, by changing the weights of the connections, seeks to modify the network response, in a direction that reduces the error. After repeating this process for a sufficiently large number of training cycles, the network will usually converge to a state of relatively small output error. At this stage, the network will have reached a certain target function. As the algorithm's name implies, the errors propagate backwards from the output to the input nodes, through the various inner layers nodes. Thus, back-propagation is used to calculate the gradient of the error of the network with respect to the network's modifiable weights. To adjust the weights properly, we use a variant of the back propagation algorithm, based on the Levenberg-Marquard Algorithm (Lourakis 2005) for non-linear optimization. In order to minimize the error, the derivative of the error function with respect to the network weights is calculated, and the weights are then adjusted to reduce the error (thus descending on the surface of the error function). For this reason, back-propagation can only be applied on networks with differentiable activation functions. Back-propagation can give to suitable local networks with quick convergence on satisfactory local error minima.

A BPNN is a feed-forward, multilayer network of standard structure, i.e., neurons are not connected with each other in the layer they belong to, but they are connected with all the neurons of the previous and subsequent layer. A BPNN has a standard structure that can be written as

$$\mathrm{N} - \mathrm{H}_1 - \mathrm{H}_2 - \cdots - \mathrm{H}_{\mathrm{NHL}} - \mathrm{M} \tag{1}$$

where N is the number of input neurons (input parameters); $\mathrm{H_i}$ is the number of neurons in the $\mathrm{i-th}$ hidden layer for $\mathrm{i} = 1, \ldots, \mathrm{NHL}$; NHL is the number of hidden layers and M is the number of output neurons (output parameters). Fig. 1 depicts an example of a BPNN composed of an input layer with 5 neurons, two hidden layers with 4 and 3 neurons respectively and an output layer with 2 neurons, i.e., a 5-4-3-2 BPNN.

Another notation for a single node (with the corresponding R-element input vector) of a hidden layer is presented in Fig. 2. For each node, the inputs signals $\mathrm{p_1}, \ \ldots, \ \mathrm{p_R}$ to that node are multiplied by the weights $\mathrm{w_{1,1}}, \ \ldots, \ \mathrm{w_{1,R}}$ of the connections to that node and the weighted values are fed to the summing junction. At that point, the dot product (W.p) of the single row matrix $\mathrm{W} = [\mathrm{w_{1,1}}, \ldots, \mathrm{w_{1,R}}]$ and the column vector $\mathrm{p} = [\mathrm{p_1}, \ \ldots, \ \mathrm{p_R}]^{\mathrm{T}}$ is generated. The threshold b (bias) is added to the dot product forming the net input n which is the argument of the transfer function f

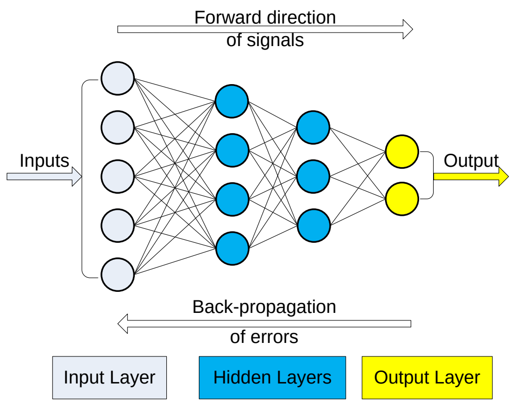


Fig. 1 A 5-4-3-2 BPNN

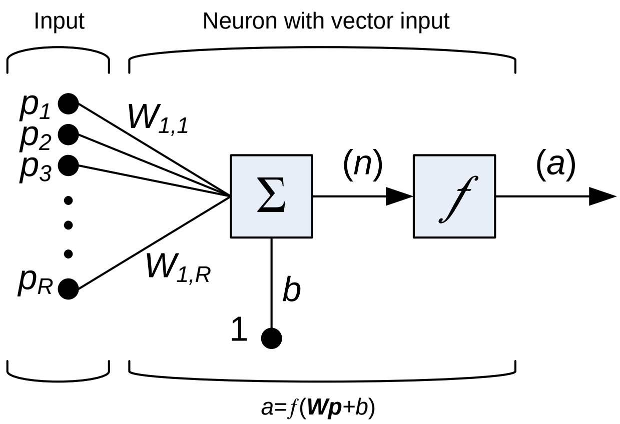


Fig. 2 A neuron with a single R-element input vector

$$n = w_{1,1}p_1 + w_{1,2}p_2 + \ ... + \ w_{1,R}p_R + b = Wp + b \tag{2}$$

2.2 Transfer functions

The choice of the transfer function may strongly influence the complexity and performance of neural networks. Transfer functions are used in ANNs as activation functions connecting the weights w_i of a neuron i to the input. Although sigmoidal transfer functions are the most commonly used, there is no a priori reason why models based on such functions should always provide optimal decision borders. Past studies (Bartlett 1998, Karlik and Olgac 2011) have proposed a large number of alternative transfer functions. In the present study the following functions are used:

The identity ('linear') transfer function

The simplest transfer function commonly used is that of the identity activation function (Fig. 3). The output of the identity function and its derivative are given by

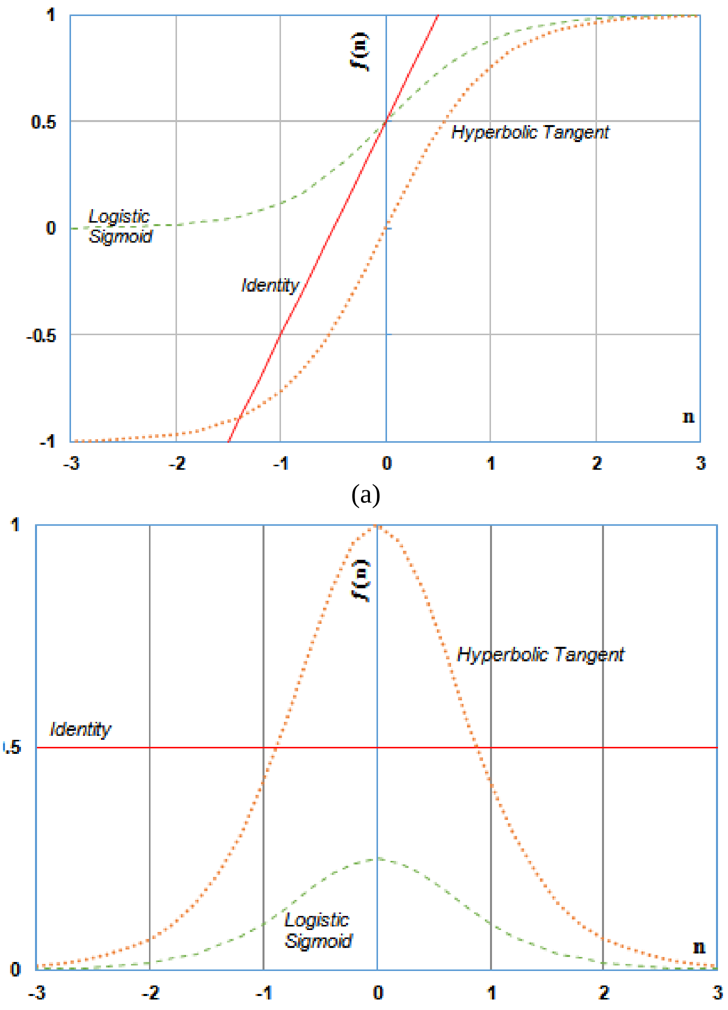


Fig. 3 Common activation functions (a) and their derivatives (b)

$$f(\mathrm{n}) = \mathrm{n} \tag{3}$$

$$f'(\mathrm{n}) = 1 \tag{4}$$

This function yields output values in the interval (-∞, +∞), while its derivative always yields output values equal to 1. It is worth mentioning that the combination of using nonlinear activation functions among the neurons of hidden units and the identity function for the output layer leads to a robust form of nonlinear regression. The network can predict continuous target values using a linear combination of signals that arise from one or more layers of nonlinear transformations of the input.

The Logistic Sigmoid Activation Function

Another function, which is often used as output activation function, is the logistic sigmoid (Fig. 3). The output of this function and its derivative are given by

$$f(\mathrm{n}) = \frac{1}{\mathrm{e}^{-\mathrm{n}} + 1} \tag{5}$$

$$f'(\mathrm{n}) = \frac{1 + \mathrm{e}^{-\mathrm{n}} + 1}{(1 + \mathrm{e}^{-\mathrm{n}})^2} \tag{6}$$

This function, yielding output values in the interval (0, +1) is suitable for binary classification problems for which the outputs values are in the interval (0, +1).

The Hyperbolic Tangent activation function

An alternative to the logistic sigmoid is the hyperbolic tangent, or tanh function (Fig. 3). The output of the hyperbolic tangent function and its derivative are given by

$$f(\mathrm{n}) = \frac{\mathrm{e}^{2\mathrm{n}} - 1}{\mathrm{e}^{2\mathrm{n}} + 1} \tag{7}$$

$$f'(\mathrm{n}) = 4\frac{\mathrm{e}^{2\mathrm{n}}}{(\mathrm{e}^{2\mathrm{n}} + 1)^2} \tag{8}$$

The tanh function is also sigmoidal (“s”-shaped). This function yields output values in the interval (-1, 1), while its derivative yields output values in the interval (0, 1). Thus strongly negative inputs to the tanh will map to negative outputs. Additionally, only zero-valued inputs are mapped to near-zero outputs. These properties make the network less likely to get “stuck” during training.

2.3 Finding the best architecture of a ANN or how to avoid the over-fitting problem

The best architecture for an ANN can be identified, given the known number of parameters for input and output, for the present application), by estimating the optimum number of hidden layers and neurons.

Finding the best architecture avoids the over-fitting problem. Over-fitting generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations as well as when the training data do not cover the entire range of the input parameters values of the problem. An extreme example of over-fitting where the number of parameters the number of parameters is equal or exceeds the observations a simple model can predict the training data by memorizing them but fails to predict new ones by not learning to generalize. In order to prevent over-fitting several techniques/algorithms and criteria have been proposed (Papadopoulos *et al.* 2012, Lamanna *et al.* 2012, Chen 2013, Giovanis and Papadopoulos 2015, Asteris *et al.* 2016) for determining the correct number of neurons with their hidden layer based mainly on the number of inputs and output parameters (Blum 1992, Boger and Guterman 1997, Berry and Linoff 1997). In the present work, we propose a simple heuristic algorithm for building a reliable and robust ANN suitable for learning and producing, during performance, the continuous mapping from the input to the output space. The steps of the proposed algorithm which can reliably predict the Surface Roughness of Electro-discharge Machined Components are as follows:

Step 1: Normalization of data: The normalization is carried out in a pre-processing stage. Pre-processing has been proved to be the most crucial step of any type problem in the field of soft computing techniques such as the artificial neural networks techniques.

Step 2: Development and training of several neural networks: The development and training of the NNs occurs with a number of hidden layers ranging from 1 to 2 and with a number of neurons ranging from nip-1 to 15, where nip correspond to input parameters. Each one of the NNs is developed and trained for a number (nf) of activation functions as well as with and without the use of data preprocessing techniques (step 1).

Step 3: Determination of mean square error: For each one of the above trained NNs the mean square error (MSE) is computed for a set of data (validation data) which have not been used during the training process (training data) of the ANNs.

Step 4: Establishment of upper and lower limits: Upper and lower limits are introduced for each one of the output parameters based on experimental or numerical data as well as reasonable estimations by the users.

Step 5: Selection of optimum architecture: The optimum architecture is the one that gives the minimum mean square error while all the computed output parameters for all the validation data are between the upper and lower limits.

It should be emphasized the importance of the limits established at Step 3 based on the user's expertise, since it is needed wide experience not only in relation to the neural networks but also to the specific field applied in order to establish reasonable estimations.

3. Results and discussion

In this section, the reliability, the effectiveness and the robustness of the above proposed algorithm, for finding of the best architecture of a BPNN, is presented through a step-by-step approach. In particular, the proposed algorithm has been applied for the prediction of the mean surface roughness of electro-discharge machined surfaces.

3.1 Experimental

ED Machining was performed on a HOSTEK SH-38GP (ZNC-P type) electro-discharge machine-tool with working voltage of 30V and open circuit voltage of 100V. Fifty four experiments were conducted in typical dielectric oil (BP250) with electrolytic copper being used as the tool electrode (anode). In particular, three different tool steels, namely AISI D2, P20 modified and premium H13 were machined (cathode), with the specimens being in the form of square plates of dimensions 70 mm×70 mm×10 mm. We varied the pulse current Ie and the pulse-on time, tp, which are considered to be the main operational parameters, over a range from roughing to near finishing. More specifically Ie was set at 5, 10, 15, 20, 25, 30 (A) and tp at 100, 300 and 500 (μs).

We performed the surface texture analysis, using a Rank Taylor-Hobson Surtronic 3+ profilometer equipped with the Talyprof® software. The cut-off length was selected at 0.8 mm whilst 20 measurements were conducted on every specimen at random directions, as it is known that EDMachining generates geometrically isotropic texture (Petropoulos *et al.* 2009). It is worth mentioning that the surface roughness parameter considered was the center-line average (mean) surface roughness, Ra. Worth mentioning that Ra is by far the most commonly used parameter in

Table 1 Experimental data/results and input and output parameters of BPNNs

Sample	Material	Input						Output	Comments *
		Material Encoding with				Pulse current I_e (A)	Pulse duration t_p (μs)	Mean Surface Roughness, R_a (μm)	
		3 parameters			1 parameter				
1	AISI D2	1	0	0	1	5	100	3.95	T
2	AISI D2	1	0	0	1	10	100	4.24	V
3	AISI D2	1	0	0	1	15	100	6.42	Test
4	AISI D2	1	0	0	1	20	100	7.95	T
5	AISI D2	1	0	0	1	25	100	7.98	V
6	AISI D2	1	0	0	1	30	100	8.12	T
7	AISI D2	1	0	0	1	5	300	5.26	T
8	AISI D2	1	0	0	1	10	300	8.27	V
9	AISI D2	1	0	0	1	15	300	9.85	Test
10	AISI D2	1	0	0	1	20	300	11.29	T
11	AISI D2	1	0	0	1	25	300	11.97	V
12	AISI D2	1	0	0	1	30	300	12.50	T
13	AISI D2	1	0	0	1	5	500	7.97	T
14	AISI D2	1	0	0	1	10	500	8.48	V
15	AISI D2	1	0	0	1	15	500	11.46	Test
16	AISI D2	1	0	0	1	20	500	13.72	T
17	AISI D2	1	0	0	1	25	500	14.15	V
18	AISI D2	1	0	0	1	30	500	14.71	T
19	AISI P20	0	1	0	2	5	100	4.39	T
20	AISI P20	0	1	0	2	10	100	4.50	V
21	AISI P20	0	1	0	2	15	100	5.61	Test
22	AISI P20	0	1	0	2	20	100	6.94	T
23	AISI P20	0	1	0	2	25	100	8.23	V
24	AISI P20	0	1	0	2	30	100	9.92	T
25	AISI P20	0	1	0	2	5	300	4.92	T
26	AISI P20	0	1	0	2	10	300	8.90	V
27	AISI P20	0	1	0	2	15	300	11.48	Test

Note: T: Training Data; V: Validation Data; Test: Test Data

surface finish measurement and for general quality control. Despite its inherent limitations, it is easy to measure and offers a good overall description of the height characteristics of a surface profile (Petropoulos *et al.* 2004).

The process parameters and the mean surface roughness values, we measured are tabulated in Table 1. Note that for the present research experiments with operational parameters reported in (Markopoulos *et al.* 2008 and Vaxevanidis *et al.* 2013) where repeated and additional experiments were performed in order to follow best practice concerning the design of experiments (Phadke 1989).

Table 1 Continued

Sample	Material	Input						Output	Comments *
		Material Encoding with				Pulse current I_e (A)	Pulse duration t_p (μs)	Mean Surface Roughness, R_a (μm)	
		3 parameters			1 parameter				
28	AISI P20	0	1	0	2	20	300	13.06	T
29	AISI P20	0	1	0	2	25	300	13.44	V
30	AISI P20	0	1	0	2	30	300	13.34	T
31	AISI P20	0	1	0	2	5	500	7.39	T
32	AISI P20	0	1	0	2	10	500	10.95	V
33	AISI P20	0	1	0	2	15	500	12.12	Test
34	AISI P20	0	1	0	2	20	500	13.39	T
35	AISI P20	0	1	0	2	25	500	14.18	V
36	AISI P20	0	1	0	2	30	500	14.65	T
37	AISI H13	0	0	1	3	5	100	5.32	T
38	AISI H13	0	0	1	3	10	100	6.01	V
39	AISI H13	0	0	1	3	15	100	6.83	Test
40	AISI H13	0	0	1	3	20	100	7.45	T
41	AISI H13	0	0	1	3	25	100	7.76	V
42	AISI H13	0	0	1	3	30	100	7.96	T
43	AISI H13	0	0	1	3	5	300	6.69	T
44	AISI H13	0	0	1	3	10	300	8.14	V
45	AISI H13	0	0	1	3	15	300	10.11	Test
46	AISI H13	0	0	1	3	20	300	11.59	T
47	AISI H13	0	0	1	3	25	300	12.20	V
48	AISI H13	0	0	1	3	30	300	12.62	T
49	AISI H13	0	0	1	3	5	500	7.68	T
50	AISI H13	0	0	1	3	10	500	8.86	V
51	AISI H13	0	0	1	3	15	500	11.37	Test
52	AISI H13	0	0	1	3	20	500	13.34	T
53	AISI H13	0	0	1	3	25	500	14.15	V
54	AISI H13	0	0	1	3	30	500	14.91	T

Note: T: Training Data; V: Validation Data; Test: Test Data

3.2 Normalization of data

As stated above, the normalization of the parameters considered in the database has a significant impact on the ANN procedure. In theory, it's not necessary to normalize numeric data. However, practice has shown that when numeric data values are normalized, neural network training is often more efficient leading to more accurate predictions. Basically, if numeric data is not normalized, and the magnitudes of two predictors are far apart, then a change in the value of a neural network weight has far more relative influence on the value with larger magnitudes.

Table 2 Training parameters of BBNN models

Parameter	Value
Training Goal	0
Training Algorithm	Levenberg-Marquardt Algorithm
Epochs	1000
Cost Function	MSE; SSE
Transfer Functions	Tansig (T); Logsig (L)
Initial Weights of Hidden Layers	1.00
Initial Weights of Bias	1.00

Note: MSE: Mean Square Error; SSE: Sum Square Error; Tansig (T): Hyperbolic Tangent Sigmoid transfer function; Logsig (L): Log-sigmoid transfer function

In the present study, during the pre-processing stage, the Min-Max (Delen *et al.* 2006) as well as the ZScore Normalization techniques have been been used. In particular, the two input parameters Pulse current (Ie) and Pulse duration (tp) as well as the output parameter Roughness (Ra) have been normalized using the above normalization methods which involve simple techniques. In particular, to avoid problems associated with low learning rates of the ANN (Iruansi *et al.* 2010) it is better to normalize the values of the parameters between an appropriate upper and lower limit value of the subject parameter. The above mentioned three parameters have been normalized for the case of MinMax normalization method for upper and lower limit value between (0, 1) and (-1, 1).

3.3 Material encoding

In developing the BPNN model, we gave special emphasis to the encoding of the experimental data. That data involved measurements from three different materials. (AISI D2, AISI P20 and AISI H13). To simulate the quality (characteristics) of the materials, we considered two cases. In the first case the representation of the materials was by means of a single input parameter with values 1, 2 and 3 for the materials AISI D2, AISI P20 and AISI H13, respectively. In the second case the representation of the material has been achieved through three input parameters with values 1, 0, 0 for the material AISI D2, 0, 1, 0 for the material AISI P20 and 0, 0, 1 for the material AISI H13.

3.4 BPNN model development

Based on the above described algorithm, 2880 BPNN models have been developed and investigated. Each one of these models was trained by means of 27 datasets (out of the total of 54, that is a 50% percentage) and the reliability of the results was validated by means of 18 datasets (33.33%) and was tested against the remaining 9 data sets (16.67% of total), by calculating the Pearson's correlation coefficient R. The parameters used for the training of NN models are summarized in Table 2. The above data sets have been selected by dividing the total data set (54 experiments) using specified indices as it is shown in the last column ("Comments") of the Table 1.

The total of the 2880 developed NN models, which have been ranked based on Pearson's

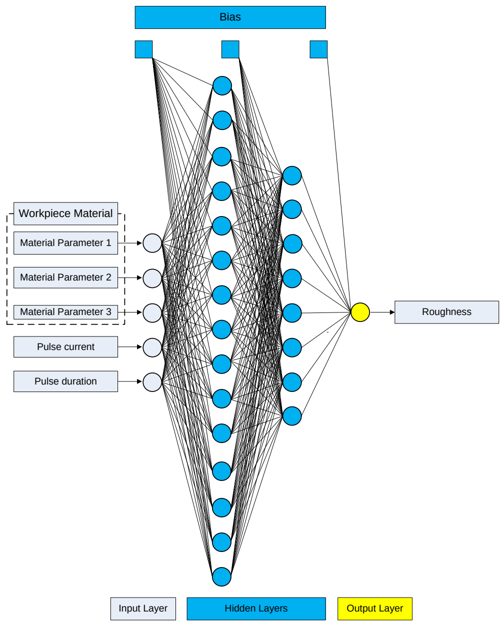


Fig. 4 The Best 5-15-8-1 BPNN based on Pearson's correlation coefficient R

correlation coefficient R and the architecture of the top twenty models are presented in Table 3. Based on these results, the optimum BPNN model is that of 5-15-8-1 (Fig. 4) with Pearson's correlation coefficient R equal to 0.99507 (see first row of Table 3 as well as Fig. 5). This network corresponds to the case of (a) architecture with two hidden layers, (b) the material has been encoded with three input parameters and (c) with ZScore normalization technique. It should be noted that the second one is that of 5-9-8-1 with Pearson's correlation coefficient R equal to 0.99505 (see second row of Table 3) which also corresponds to a NN with two hidden layers and has been derived using the MinMax normalization technique for the case of upper and lower limit value between (-1, 1).

Fig. 6 presents the experimental values in comparison to the predicted values by the optimum BPNN model of 5-15-8-1. It is clear that the mean surface roughness of electro-discharge machined surfaces predicted from the multilayer feed-forward neural network, are very close to the experimental results.

It is also worth mentioning that

• The majority of the top twenty models (19 from 20) corresponds to models with two hidden layers,

• The way of encoding the material is a crucial parameter; in 17 from the top twenty models the material has been modelled using three inputs parameters,

• The majority of the top twenty models (17 from 20), including the optimum one, corresponds to models where the data have been normalized,

• The total of the top twenty models presented in Table 3 have been trained under a number of epochs range between 39 and 168, which means that the developed multilayer feed-forward neural network models can predict the mean surface roughness of electro-discharge machined surfaces with smaller error rates and less computational effort compared with those available in the literature models.

Table 3 Ranking of the top twenty best architectures of BPNNs based on Pearson's correlation coefficient R

Ranking	Material Encoding	Preprocess	Cost Function	Architecture (Code)	Training Functions	Pearson's R	Number of Epochs
1	3	ZScore	SSE	5-15-8-1	T-T-T	0.99507	91
2	3	MinMax (-1, 1)	MSE	5-9-8-1	T-L-T	0.99505	75
3	3	ZScore	SSE	5-12-12-1	T-L-T	0.99472	114
4	3	No	MSE	5-14-10-1	T-L-T	0.99450	67
5	1	MinMax (-1, 1)	MSE	3-13-5-1	T-T-T	0.99448	49
6	3	MinMax (0, 1)	MSE	5-14-1	T-T	0.99438	112
7	3	MinMax (0, 1)	SSE	5-11-6-1	T-T-T	0.99435	51
8	3	ZScore	SSE	5-9-5-1	T-L-T	0.99415	82
9	3	ZScore	SSE	5-13-7-1	T-T-T	0.99395	39
10	3	No	MSE	5-13-11-1	T-T-T	0.99363	79
11	1	MinMax (-1, 1)	SSE	3-10-5-1	T-T-T	0.99357	54
12	3	No	SSE	5-9-9-1	T-T-T	0.99353	117
13	3	MinMax (-1, 1)	SSE	5-12-6-1	T-T-T	0.99346	75
14	3	MinMax (0, 1)	MSE	5-15-6-1	T-T-T	0.99326	76
15	3	MinMax (-1, 1)	SSE	5-12-11-1	T-T-T	0.99310	168
16	1	ZScore	SSE	3-10-8-1	T-T-T	0.99306	89
17	3	MinMax (0, 1)	MSE	5-9-5-1	T-T-T	0.99293	78
18	3	MinMax (0, 1)	MSE	5-15-9-1	T-T-T	0.99293	114
19	3	MinMax (0, 1)	MSE	5-10-10-1	T-L-T	0.99290	76
20	3	ZScore	SSE	5-15-8-1	T-T-T	0.99507	91

Note: Material Encoding: 3 means that 3 inputs parameters have been used for the encoding of the material encoding while Material Encoding: 1 means that only1 input parameter has been used for the encoding of the material

In order to be useful, a proposed NN architecture should be accompanied by the (quantitative) values of weights. In such a case, the NN model can be readily implemented in an MS-Excel spreadsheet, thus available to anyone interested in the problem of modelling. To this end, the final values of weights and biases of the optimum NN architecture are explicitly reported in Fig. 7.

4. Conclusions

In this paper, the artificial neural networks method was assessed by investigating its accuracy in predicting the mean surface roughness of electro-discharge machined surfaces. In particular, a novel heuristic algorithm was proposed to find the optimal architecture out of a set of multi layered feed-forward back-propagation neural networks. Using this algorithm a ranking list of the best architectures of neural network models based on the Pearson's correlation coefficient R was selected. The mean surface roughness of electro-discharge machined surfaces predicted from the multilayer feed-forward neural network, are very close to the experimental results as confirmed by correlation coefficient R. In conclusion, mean surface roughness of electro-discharge machined

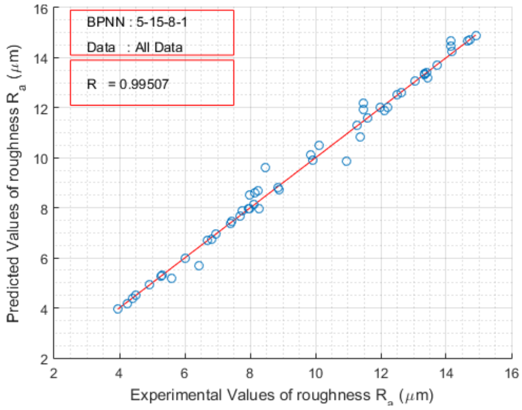


Fig. 5 Pearson's correlation coefficient R of the experimental and predicted Roughness, Ra for the best with two hidden layers BPNN (5-15-8-1)

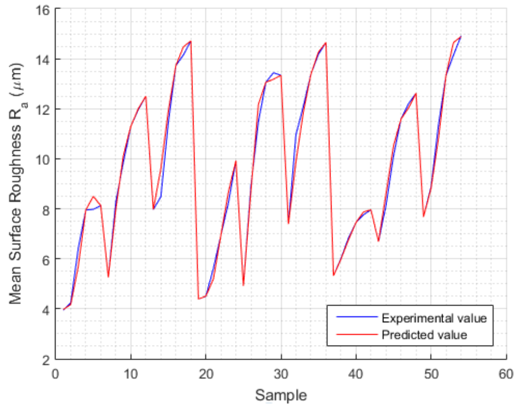
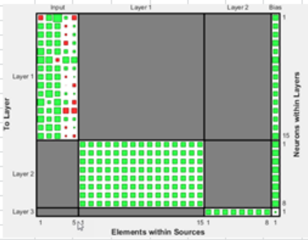


Fig. 6 Experimental vs Predicted values of Surface Roughness

iw{1.1}

-0.737	1.205	2.480	0.526	-0.642
1.104	1.030	0.969	-0.145	0.168
1.126	1.037	0.938	-0.128	0.233
0.863	1.066	0.994	-0.432	0.322
1.384	0.732	1.163	0.423	0.164
1.765	1.369	-0.286	1.095	0.868
0.859	1.115	1.096	0.312	0.215
0.932	1.256	1.012	-0.151	0.024
1.165	1.016	1.036	-0.092	-0.336
0.841	1.566	0.796	0.178	0.154
1.426	0.385	1.439	0.603	-0.483
0.954	0.818	1.363	-0.628	-0.589
1.361	0.727	1.181	0.395	0.176
0.931	1.274	0.985	-0.145	0.071
1.122	0.769	1.010	-0.099	-0.151



b{1.1}
1.052
0.897
0.899
1.077
0.720
1.152
0.929
0.800
0.783
0.797
0.750
0.865
0.732
0.810
1.100

iw{{2.1}}

0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791
0.911	0.789	0.790	0.797	0.794	0.858	0.790	0.790	0.796	0.790	0.785	0.797	0.794	0.790	0.791

b{{2.1}
1.201
1.201
1.201
1.201
1.201
1.201
1.201
1.201

iw{3.2}

0.584	0.584	0.584	0.584	0.584	0.584	0.584	0.584

b{3.1}
0.119

iw{1.1}= Weights values for Input Layers
iw{2.1}= Weights values for first hidden Layers
iw{3.2}= Weights values for second hidden Layers
b{1.1}= Bias values for first hidden layer
b{2.1}= Bias values for second hidden layer
b{3.1}= Bias values for output layer

Fig. 7 Final Weights and Bias Values of the optimum BPNN model 5-15-8-1 surfaces can be predicted by multilayer feed-forward neural network model with smaller error rates and less computational effort.

As delineated by the authors of this paper, a more reliable NN architecture (with respect to the presented herein) can be developed for the prediction of the mean surface roughness of electro-discharge machined surfaces. To this end a set of additional optimization functions will be used for the training of the NNs as well all a set of different computational units will be used to find the best architecture.

Acknowledgments

The research was performed within the framework of the Master's Program in Applied Computational Structural Engineering (ACSE), which was partially financed by the Research Committee of the School of Pedagogical & Technological Education, Athens, Greece.

References

Adeli, H. (2001), "Neural networks in civil engineering: 1989-2000", *Comput.-Aid. Civil Infrastr. Eng.*, **16**(2), 126-142.

Al-Ghamdi, K.A. and Aspinwall, E. (2014), "Modelling an EDM process using multilayer perceptron network, RSM, and high-order polynomial", *Adv. Mech. Eng.*, **6**, 791242.

Al-Ghamdi, K. and Taylan, O. (2015), "A comparative study on modelling material removal rate by ANFIS and polynomial methods in electrical discharge machining process", *Comput. Industr. Eng.*, **79**, 27-41.

Asteris, P.G. and Plevris, V. (2013), "Neural network approximation of the masonry failure under biaxial compressive stress", *Proceedings of the 3rd South-East European Conference on Computational Mechanics*, 584-598.

Asteris, P.G., Tsaris, A.K., Cavaleri, L., Repapis, C.C., Papalou, A., Di Trapani, F. and Karypidis, D.F. (2016), "Prediction of the fundamental period of infilled rc frame structures using artificial neural networks", *Comput. Intell. Neurosci.*, 5104907.

Asteris, P.G. and Plevris, V. (2016), "Anisotropic masonry failure criterion using artificial neural networks", *Neur. Comput. Appl.*, **28**(8), 2207-2229.

Asteris, P.G., Kolovos, K.G., Douvika, M.G. and Roinos, K. (2016), "Prediction of self-compacting concrete strength using artificial neural networks", *Eur. J. Environ. Civil Eng.*, **20**, s102-s122.

Asteris, P.G., Roussis, P.C. and Douvika, M.G. (2017), "Feed-forward neural network prediction of the mechanical properties of sandcrete materials", *Sens.*, **17**(6), 1344.

Asteris, P.G. and Kolovos, K.G. (2017), "Self-compacting concrete strength prediction using surrogate models", *Neur. Comput. Appl.*, 1-16.

Bartlett, P.L. (1998), "The sample complexity of pattern classification with neural networks: The size of the weights is more important than the size of the network", *IEEE Trans. Informat. Theor.*, **44**(2), 525-536.

Berry, M.J.A. and Linoff, G. (1997), *Data Mining Techniques*, John Wiley & Sons, New York, U.S.A.

Blum, A. (1992), *Neural Networks in C++*, Wiley, New York, U.S.A.

Boger, Z. and Guterman, H. (1997), "Knowledge extraction from artificial neural network models", *Proceedings of the IEEE Systems, Man, and Cybernetics Conference*, Orlando, Florida, U.S.A.

Chen, Z. (2013), *An Overview of Bayesian Methods for Neural Spike Train Analysis*, Computational Intelligence and Neuroscience.

Das, M.K., Kumar, K., Barman, T.K. and Sahoo, P. (2014), "Prediction of surface roughness in edm using response surface methodology and artificial neural network", *J. Manufact. Technol. Res.*, **6**(3-4), 93-112.

Delen, D., Sharda, R. and Bessonov, M. (2006), "Identifying significant predictors of injury severity in

traffic accidents using a series of artificial neural networks", *Accid. Analys. Prevent.*, **38**(3), 434-444.
Dini, G. (1997), "Literature database on applications of artificial intelligence methods in manufacturing engineering", *Ann. CIRP*, **46**(2), 681-690.
Giovanis, D.G. and Papadopoulos, V. (2015), "Spectral representation-based neural network assisted stochastic structural mechanics", *Eng. Struct.*, **84**, 382-394.
Hornik, K., Stinchcombe, M. and White, H. (1989), "Multilayer feedforward networks are universal approximators", *Neur. Network.*, **2**(5), 359-366.
Huang, J.C., Chang, H., Kuo, C.G., Li, J.F. and You, Y.C. (2015), "Prediction surface morphology of nanostructure fabricated by nano-oxidation technology", *Mater.*, **8**(12), 8437-8451.
Iruansi, O., Guadagnini, M., Pilakoutas, K. and Neocleous, K. (2010), "Predicting the shear strength of RC beams without stirrups using bayesian neural network", *Proceedings of the 4th International Workshop on Reliable Engineering Computing (REC 2010)*.
Karlik, B. and Olgac, A.V. (2011), "Performance analysis of various activation functions in generalized MLP architectures of neural networks", *J. Artific. Intell. Exp. Syst.*, **1**(4), 111-122.
Kumar, S., Batish, A., Singh, R. and Singh, T.P. (2014), "A hybrid Taguchi-artificial neural network approach to predict surface roughness during electric discharge machining of titanium alloys", *J. Mech. Sci. Technol.*, **28**(7), 2831-2844.
Lamanna, J., Malgaroli, A., Cerutti, S. and Signorini, M.G. (2012), "Detection of fractal behavior in temporal series of synaptic quantal release events: A feasibility study", *Comput. Intell. Neurosci.*, **3**.
Lourakis, M.I.A. (2005), *A Brief Description of the Levenberg-Marquardt Algorithm Implemented by Levmar, Technical Report*, Institute of Computer Science, Foundation for Research and Technology, Hellas.
Mansouri, I. and Kisi, O. (2015), "Prediction of debonding strength for masonry elements retrofitted with FRP composites using neuro fuzzy and neural network approaches", *Compos. Part B: Eng.*, **70**, 247-255.
Mansouri, I., Ozbakkaloglu, T., Kisi, O. and Xie, T. (2016), "Predicting behavior of FRP-confined concrete using neuro fuzzy, neural network, multivariate adaptive regression splines and M5 model tree techniques", *Mater. Struct./Materiaux Constr.*, **49**(10), 4319-4334.
Mansouri, I., Gholampour, A., Kisi, O. and Ozbakkaloglu, T. (2016), "Evaluation of peak and residual conditions of actively confined concrete using neuro-fuzzy and neural computing techniques", *Neur. Comput. Appl.*, 1-16.
Mansouri, I., Hu, J.W. and Kişi, O. (2016), "Novel predictive model of the debonding strength for masonry members retrofitted with FRP", *Appl. Sci.*, **6**(11), 337.
Markopoulos, A.P., Manolakos, D.E. and Vaxevanidis, N.M. (2008), "Artificial neural network models for the prediction of surface roughness in electrical discharge machining", *J. Intell. Manufact.*, **19**(3), 283-292.
Moghaddam, M.A. and Kolahan, F. (2015), "An optimised back propagation neural network approach and simulated annealing algorithm towards optimisation of EDM process parameters", *J. Manufact. Res.*, **10**(3), 215-236.
Papadopoulos, V., Giovanis, D.G., Lagaros, N.D. and Papadrakakis, M. (2012), "Accelerated subset simulation with neural networks for reliability analysis", *Comput. Meth. Appl. Mech. Eng.*, **223-224**, 70-80.
Pattnaik, S., Karunakar, D.B. and Jha, P.K. (2014), "A prediction model for the lost wax process through fuzzy-based artificial neural network", *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, **228**(7), 1259-1271.
Petropoulos, G., Vaxevanidis, N.M. and Pandazaras, C. (2004), "Modeling of surface finish in electro-discharge machining based upon statistical multi-parameter analysis", *J. Mater. Process. Technol.*, **155**, 1247-1251.
Petropoulos, G., Vaxevanidis, N., Iakovou, A. and David, K. (2006), "Multi-parameter modeling of surface texture in EDMachining using the design of experiments methodology", *InMater. Sci. For.*, **526**, 157-162.
Petropoulos, G.P., Vaxevanidis, N.M., Radovanovic, M. and Zoler, C. (2009), "Morphological: Functional aspects of electro-discharge machined surface textures", *Strojniški Vestnik*, **55**(2), 95-103.

Phadke, M.S. (1989), *Quality Engineering Using Design of Experiments*, In Quality Control, Robust Design, and the Taguchi Method, Springer US, 31-50.

Plevris, V. and Asteris, P.G. (2014), “Modeling of masonry compressive failure using neural networks”, *Proceedings of the 1st International Conference on Engineering and Applied Sciences Optimization*, 2843-2861.

Plevris, V. and Asteris, P.G. (2014), “Modeling of masonry failure surface under biaxial compressive stress using neural networks”, *Constr. Build. Mater.*, **55**, 447-461.

Plevris, V. and Asteris, P. (2015), “Anisotropic failure criterion for brittle materials using artificial neural networks”, *Proceedings of the 5th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering*, 2259-2272.

Porwal, R.K., Yadava, V. and Ramkumar, J. (2014), “Neural network based modelling and GRA coupled PCA optimization of hole sinking electro discharge micromachining”, *J. Manufact. Mater. Mech. Eng.*, **4**(1), 1-21.

Pradhan, M.K. and Das, R. (2015), “Application of a general regression neural network for predicting radial overcut in electrical discharge machining of AISI D2 tool steel”, *J. Mach. Mach. Mater.*, **17**(3-4), 355-369.

Pramanick, A., Saha, N., Dey, P.P. and Das, P.K. (2014), “Wire EDM process modeling with artificial neural network and optimization by grey entropy-based taguchi technique for machining pure zirconium diboride”, *J. Manufact. Technol. Res.*, **5**(3-4), 99-116.

Rahman Khan, M.A., Rahman, M.M. and Kadirgama, K. (2014), “Neural network modeling and analysis for surface characteristics in electrical discharge machining”, *Proc. Eng.*, **90**, 631-636.

Sarkheyli, A., Zain, A.M. and Sharif, S. (2015), “A multi-performance prediction model based on ANFIS and new modified-GA for machining processes”, *J. Intell. Manufact.*, **26**(4), 703-716.

Shrivastava, P.K. and Dubey, A.K. (2014), “Electrical discharge machining-based hybrid machining processes: A review”, *Part B*: *J. Eng. Manufact.*, **228**(6), 799-825.

Tang, L. and Guo, Y.F. (2014), “Electrical discharge precision machining parameters optimization investigation on S-03 special stainless steel”, *J. Adv. Manufact. Technol.*, **70**(5-8), 1369-1376.

Vaxevanidis, N.M., Kechagias, J.D., Fountas, N.A. and Manolakos, D.E. (2014), “Evaluation of machinability in turning of engineering alloys by applying artificial neural networks”, *Open Constr. Build. Technol. J.*, **8**(1), 389-399.

Vaxevanidis, N.M., Fountas, N., Tsakiris, E., Kalogeropoulos G. and Sideris, J. (2013), “Multi parameter analysis and modeling of surface finish in electro-discharge machining of tool steels”, *Nonconvent. Technol. Rev.*, **27**(3), 87-90.

Wang, K., Gelgele, H.L., Wang, Y., Yuan, Q. and Fang, M. (2003), “A hybrid intelligent method for modelling the EDM process”, *J. Mach. Tool. Manufact.*, **43**, 995-999.

Wang, G., Zhou, H., Wang, Y. and Yuan, X. (2014), “Modeling surface roughness based on artificial neural network in mould polishing process”, *Proceedings of the IEEE International Conference on Mechatronics and Automation*, 799-804.

Xu, Y. and Gao, T. (2016), “Optimizing thermal-elastic properties of C/C-SiC composites using a hybrid approach and PSO algorithm”, *Mater.*, **9**(4), 222.

Zhang, W., Bao, Z., Jiang, S. and He, J. (2016), “An artificial neural network-based algorithm for evaluation of fatigue crack propagation considering nonlinear damage accumulation”, *Mater.*, **9**(6), 483.

CC