Pergamon International Journal of Machine Tools & Manufacture 41 (2001) 1385–1403

INTERNATIONAL JOURNAL OF MACHINE TOOLS & MANUFACTURE DESIGN, RESEARCH AND APPLICATION

Predictions on surface finish in electrical discharge machining based upon neural network models

Kuo-Ming Tsai, Pei-Jen Wang *

Department of Power Mechanical Engineering, National Tsing Hua University, 101, Sec. II, Kuang Fu Rd, Hsinchu, 30013, Taiwan, ROC

Received 18 July 2000; accepted 9 February 2001

Abstract

Predictions on the surface finish of work-pieces in electrical discharge machining (EDM) based upon physical or empirical models have been reported in the past years. However, when the change of electrode polarity has been considered, very few models have given reliable predictions. In this study, the comparisons on predictions of surface finish for various work materials with the change of electrode polarity based upon six different neural-networks models and a neuro-fuzzy network model have been illustrated. The neural-network models are the Logistic Sigmoid Multi-layered Perceptron (LOGMLP), the Hyperbolic Tangent Sigmoid Multi-layered Perceptron (TANMLP), the Fast Error Back-propagation Hyperbolic Tangent Multi-layered Perceptron (Error TANMLP), the Radial Basis Function Networks (RBFN), the Adaptive Hyperbolic Tangent Sigmoid Multi-layered Perceptron, and the Adaptive Radial Basis Function Networks. The neuro-fuzzy network is the Adaptive Neuro-Fuzzy Inference System (ANFIS). Being trained by experimental data initially screened by the Design of Experiment (DOE) method, the parameters of the above models have been optimally determined for predictions. Based upon the conclusive results from the comparisons on checking errors among these prediction models, the TANMLP, RBFN, Adaptive RBFN, and ANFIS model have shown consistent results. Also, it is concluded that the further experimental results have agreed to the predictions based upon the above four models.

1. Intrduction

Electrical discharge machining (EDM), is a nontraditional machining process for metals removing based upon the fundamental fact that negligible tool force is generated during the machining

* Corresponding author. Tel.: +886 3 571 9034; fax: +886 3 572 2840.
E-mail address: pjwang@pme.nthu.edu.tw (P.-J. Wang).

0890-6955/01/$ - see front matter
PII: S0890-6955(01)00028-1

process. The removal of metals in the process is characterized by the erosive effects from a series of electrical sparks generated between tool and work materials with constant electric field emerged in dielectric environment. The EDM process is typically used for manufacturing cutting tools, punch dies, and other difficult-to-cut parts. Although the process has been accepted as the standard machining process in the tools, dies, and molds industry, the process is yet treated as the so-called "know-how" process today. That means, the tuning of EDM process variables for obtaining process efficiency and part accuracy has been empirical. Even though the up-to-date computer technology has been applied on the machine controller, the EDM process is still one of the expertise-demanding processes in the industry. From the literatures, the comprehensive mechanism of metal erosion during sparking is still debatable although the basic physical laws have been laid for many years. On the other hand, complex thermal conduction behaviors have been widely accepted as the principal mechanism of metal erosion based upon ad-hoc engineering approach. This explains why the models for correlating the process variables and surface finish are hard to be established accurately.

In the past decade, neural networks have been shown to be the highly flexible modeling tools with capabilities on learning the mathematical mapping between input variables and output features for nonlinear systems [1,2]. Also, the superior performances of neural networks for modeling machining processes have been published elsewhere [3–20]. In these, multi-layer Perceptrons based on back-propagation (BP) technique have been employed for monitoring and modeling the reported processes. For example, Rangwala and Dornfeld [3], Masory [4], Tansel et al. [5], and Tarng et al. [6,7] have employed BP or Adaptive Resonance Theory (ART2) on the neural networks for monitoring tool wear and breakage in turning or drilling processes. On the other hand, Tansel et al. [8,9], and Lee et al. [10] have also adopted BP or ART2-A on the neural networks for detecting and suppressing tool chatter in turning or drilling processes. In order to increase the reliability of detection method, Li et al. [11] have developed parallel multi-ART2 neural networks for identifying tool failure due to chatter in the turning operation. The percentage of effective detection has been increased from 80.4% on single-ART2 case to 96.4% on parallel multi-ART2 case. In addition, Cariapa et al. [12], Tarng et al. [13,14], Liao and Lin [15], Lee et al. [16], and Ko and Cho [17] have applied various neural networks for modeling and predicting the machining processes. In the EDM process, both Kao and Tarng [18] and Liu and Tarng [19] have employed feed-forward neural networks with hyperbolic tangent functions and abductive networks for the on-line recognition of pulse-types. Based on their results, discharge pulses have been identified and then employed for controlling the EDM machine. Meanwhile, Katz and Naude [20] have adopted back propagation errors on neural networks for improving the geometric shape of EDM products based upon coupling feature design in the EDM process. Never the less, the effects of the change of electrode polarity have all been intentionally neglected without proper explanations.

The objective of this paper is to establish surface finish models based upon the various neural-networks for predicting the surface finish with the effects of change of electrode polarity in the EDM process. First of all, pertinent process variables affecting surface finish, such as the polarity of electrode, the discharge time, the peak current, and the materials of both tool and work, have been screened by making use of the Taguchi Method on Design of Experiment [21]. The experimental data based upon DOE have later been used for training the various neural-networks models. Finally, more experimental verifications on the established models have been conducted; and, comparisons among the models have been analyzed.

2. Neural networks

In the past decades, numerous studies have been reported on the development of neural networks based on different architectures [1,2,22–24]. Basically, one can characterize neural networks by its important features, such as the architecture, the activation functions, and the learning algorithms [2]. Each category of the neural networks would have its own input–output characteristics, and therefore it can only be applied for modeling some specific processes. On the other hand, if we look at the recent development on fuzzy theory, one also can observe that a fuzzy inference system could map a single given input to multi-outputs in a nonlinear domain similar to characteristics of the neural networks. Comparing to the neural networks, the fuzzy inference system mainly consists of membership functions, fuzzy logic operators, and prescribed if–then rules as described in the literatures. In 1993, Jang [25] first introduced the Adaptive Neuro-Fuzzy Inference System (ANFIS), which was reported as a very efficient system for solving the ill-defined equations involving the automatic elicitation of knowledge expressed only by the if–then rules. Based on our experiences, the ANFIS system seems can only be applied to the cases with seven inputs (or less) and one output. Therefore, it is still a valid candidate for the models in this paper.

As a result, six neural networks and a neuro-fuzzy network are employed for modeling the surface finish in the EDM process in this study. All the models are verbally defined as follows with equivalent abbreviations.

1. Logistic Sigmoid Multi-layered Perceptron (LOGMLP).
2. Hyperbolic Tangent Sigmoid Multi-layered Perceptron (TANMLP).
3. Radial Basis Function Network (RBFN).
4. Fast Error Back-Propagation Multi-layered Perceptron with Hyperbolic Tangent Functions (Error TANMLP).
5. Hyperbolic Tangent Sigmoid Multi-layered Perceptron with Adaptive Learning Rate (Adaptive TANMLP).
6. Radial Basis Function Network with Adaptive Learning Rate (Adaptive RBFN).
7. Adaptive Neuro-Fuzzy Inference System (ANFIS).

2.1. Architectures

As explained in the previous section, neural networks are in general categorized by their architecture. In Figs. 1 and 2, the architecture of the Multi-layered Perceptron Networks and the RBFN are shown schematically. In the two figures it should be noted that the number of hidden layers is critical for the convergence rate at the stage of training the network parameters. Empirically speaking, one hidden layer should be sufficient in the Multi-layered Perceptron Networks because the number of neurons is typically assumed to be dominant in the networks. In other words, the number of neurons must be determined by an optimization method. Furthermore, the architecture of the ANFIS has been selected based upon a first-order Sugeno fuzzy inference model, as shown in Fig. 3. The advantages of the Sugeno inference mechanism are the high computational efficiency, the built-in optimal and adaptive sheme, and the guaranteed continuity on the output

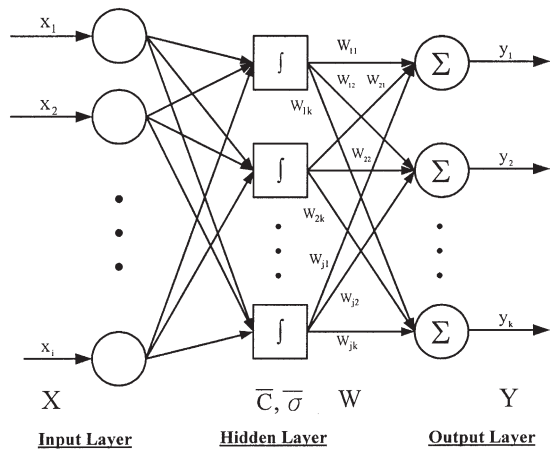

Fig. 1. Architecture of the Multi-layered Perceptron Networks.

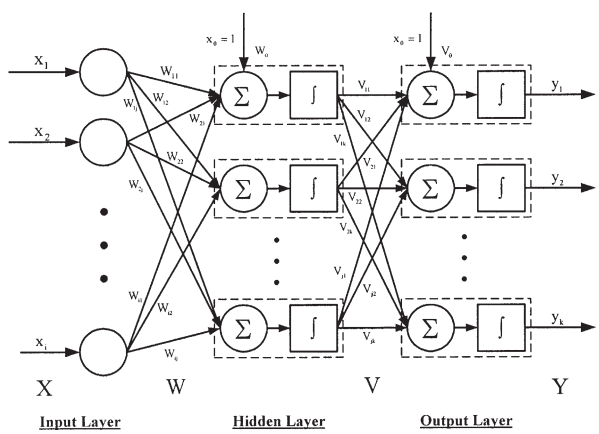

Fig. 2. Architecture of the RBFN.

surfaces [26]. In addition, a hybrid-learning algorithm for identifying of the quasi-optimal membership functions and other rule-based parameters has also been employed in this study.

2.2. Activation functions

For the neural networks shown in Figs. 1 and 2, there are many neurons in the hidden layers. The connections among the neurons are made by signal links designated by corresponding weightings. Each individual neuron is represented an internal state, namely the activation, which is functionally dependent of the inputs. In general, the Sigmoid functions (S-shaped curves), such as logistic functions and hyperbolic tangent functions, are adopted for representing the activation. In the networks, a neuron sends its activation to the other neurons for information exchange via signal links. In this paper, several different functions for activation have been employed for comparisons. And, they are described in the following paragraphs.

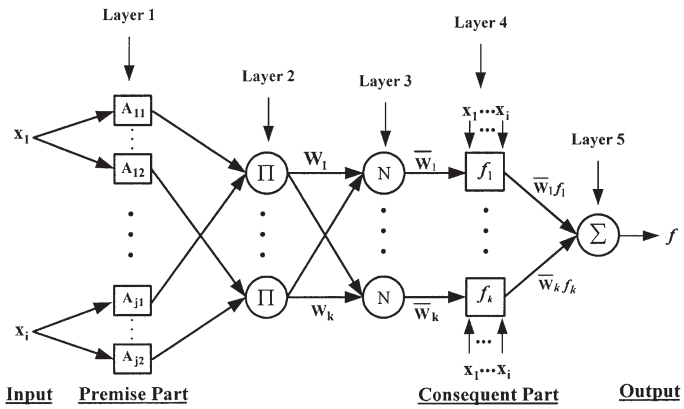

Fig. 3. Architecture of the ANFIS networks with multi-input Sugeno fuzzy model plus multi-rule; i.e. two membership functions for each input.

The activation function for the LOGMLP model is a continuous logistic function given as follows:

$$f(\text{net})=\frac{1}{1+\exp(-\lambda\text{net})} \tag{1}$$

where $\lambda>0$ is proportional to the gain which determines the steepest direction of the continuous function f(net) near net=0. Similarly, for the TANMLP model and the Error TANMLP model, continuous hyperbolic tangent functions are employed and defined as follows:

$$f(\text{net})=\tanh\left(\frac{\lambda\text{net}}{2}\right)=\frac{1-\exp(-\lambda\text{net})}{1+\exp(-\lambda\text{net})} \tag{2}$$

where λ is the same as in Eq. (1). For the RBFN model, the Gaussian distribution functions are used and defined as follows:

$$f(\text{net})=f(X;\, C_i;\, \sigma_i)=\exp\left(-\frac{|X-C_i|^2}{2\sigma_i^2}\right) \tag{3}$$

where the C_i is the center of the Gaussian distribution, and σ_i is the standard deviation.

For the ANFIS model, one of the membership functions has been chosen as the Gaussian functions, but the others have been the bell-shape functions defined as follows:

$$f(\text{net})=f(X;\, a_i,\, b_i,\, c_1)=\frac{1}{1+\left|\frac{X-c_i}{a_i}\right|^{2b_i}} \tag{4}$$

where the $\{a_i,\, b_i,\, c_i\}$ is a parameter set for the function. Because the parameter set determines the X coordinates of the two corners represented by the bell-shape functions, the parameter b_i is usually positive. If b_i is negative, then the shape of the functions becomes upside-down.

2.3. Algorithms

At the stage of training the neural networks, it is critical to select an appropriate algorithm because the efficiency and the convergence of the training are the primary issue at this stage. That means the algorithm is for determining the weightings in order to accomplish the desired mapping between the inputs and the outputs. Based on a least-square approach, the quadratic error function E between the actual outputs and the network outputs is expressed by:

$$E=\sum_p E_p=\sum_p \frac{1}{2}(T_p-Y_p)^2 \tag{5}$$

where T_p is target values and Y_p is outputs of neural networks.

In this paper, only supervised algorithms, such as the delta learning rule (or namely the gradient descent) with momentum, the fast error back-propagation learning rule with momentum, and hybrid of the delta learning and the least square estimator have been employed. In Figs. 4–6, the three algorithms are shown, respectively. Meanwhile, the ANFIS model has used a special hybrid-learning algorithm for updating its parameters. Considering the convergence criteria, both the least-squares method and the back-propagation gradient descent method have been employed for the linear and the nonlinear parameters, respectively. In all the above algorithms, an error measure for final check, which is a normalized root-mean-square of error (RMSE), is defined as follows:

$$\mathrm{RMSE}=\frac{1}{\mathrm{Length}(T_p-Y_p)}\sqrt{\sum_p \frac{1}{2}(T_p-Y_p)^2} \tag{6}$$

where T_p is the actual target vector (i.e. experimental values) and Y_p is the predicted vector

Forward pass:

Hidden layer:

$$net_j = w_0 + \sum_i w_{ij} x_i$$

$$H_j = f(net_j)$$

Output layer:

$$net_j = v_0 + \sum_j v_{jk} H_j$$

$$Y_k = f(net_j)$$

Error Function:

$$E = \sum_p E_p = \sum_p \frac{1}{2}(T_p - Y_p)^2$$

Back propagation:

$$\Delta V_j = -\eta \frac{\partial E}{\partial V_j} + \alpha V_{old}$$

$$V_{new} = V_{old} + \Delta V_j$$

where η is the learning rate, $\eta \in (0,1]$, and

α is the momentum coefficient, $\alpha \in [0,1]$.

$$\Delta W_i = -\eta \frac{\partial E}{\partial W_i} + \alpha W_{old}$$

$$W_{new} = W_{old} + \Delta W_i$$

Fig. 4. Mathematical algorithms based upon the Gradient Descent Method adopted in the LOGMLP, TANMLP, and Adaptive TANMLP networks.

Forward pass:

Hidden layer:

$$net_j = w_0 + \sum_i w_{ij} x_i$$

$$H_j = f(net_j)$$

Output layer:

$$net_k = v_0 + \sum_j v_{jk} H_j$$

$$Y_k = f(net_k)$$

Error Function:

$$E = \sum_p E_p = \sum_p \frac{1}{2}(T_p - Y_p)^2$$

$$\lambda = \exp(-\mu / E^2)$$

Back propagation:

$$E_o(\lambda) = \lambda E_p + (1-\lambda)\tanh(\beta E_p)$$

$$V_{new} = V_{old} + \eta E_o(\lambda) H_j$$

where λ is a weight factor that define a variety of alternative objective functions between the two extremes corresponding to $\lambda = 0$ and $\lambda = 1$, $\lambda \in [0,1]$, and η is the learning rate, $\eta \in (0,1]$.

$$E_j(\lambda) = (1 - H_j^2) \sum E_o(\lambda) W_{ij}$$

$$W_{new} = W_{old} + \eta E_j(\lambda) X$$

Fig. 5. Mathematical algorithm based upon the Fast Error Back-Propagation adopted in the Error TANMLP networks.

(training values). As for the adaptive algorithms, the values of the learning coefficients have to be adequately increased when the RMSE of current epoch is smaller than the RMSE of previous epoch. Otherwise, the values have to be adaptively decreased when the RMSE of current epoch is larger than the RMSE of previous epoch.

3. Experimental verifications

A schematic drawing of the experimental apparatus and a photograph of the EDM machine attached with a personal computer for data-acquisition are shown in Fig. 7. All the experiments have been conducted on a Model Mold Maker III CNC EDM machine, made by Sodick Inc. in Japan. The EDM machine was attached with a MARK XI pulse charge generator and a computer-based controller to generate rectangular-shaped current pulses during discharging. Throughout the experiments, the dielectric fluid has been the SPE oil produced by Castrol Inc. In particular, for better control of the environment, the dielectric fluid was kept in a stainless steel container during each run of the experiments. The surface finish data were later measured by a Hommel Tester T1000 profilemeter, made by Hommelwerke Inc. in Germany.

In this study, three different virgin metals were employed for the experimentation. While copper was used as the tool (upper electrode), aluminum and iron were used as the work (lower electrode). In all experiments, the pertinent process parameters and their levels for each set of the experiments are listed in Table 1. Also, the physical characteristics together with the mechanical dimensions

Forward pass:

Hidden layer:

$$H_j = f(X_i)$$

Output layer:

$$Y_k = \sum_j w_{jk} H_j$$

Error Function:

$$E = \sum_p E_p = \sum_p \frac{1}{2}(T_p - Y_p)^2$$

Back propagation:

Gradient Descent:

$$\Delta C_i = -\eta \frac{\partial E}{\partial C_i} + \alpha C_{old}$$

where η is the learning rate, $\eta \in (0,1]$, and

α is the momentum coefficient, $\alpha \in [0,1]$.

$$C_{new} = C_{old} + \Delta C_i$$

$$\Delta \sigma_i = -\eta \frac{\partial E}{\partial \sigma_i} + \alpha \sigma_{old}$$

$$\sigma_{new} = \sigma_{old} + \Delta \sigma_i$$

Least Square Estimator:

$$W_{new} = (H_j{}^t H_j)^{-1} H_j{}^t Y$$

Fig. 6. Mathematical algorithm based upon both the Gradient Descent Method and the Least-Square Estimator adopted in the RBFN and Adaptive RBFN networks.

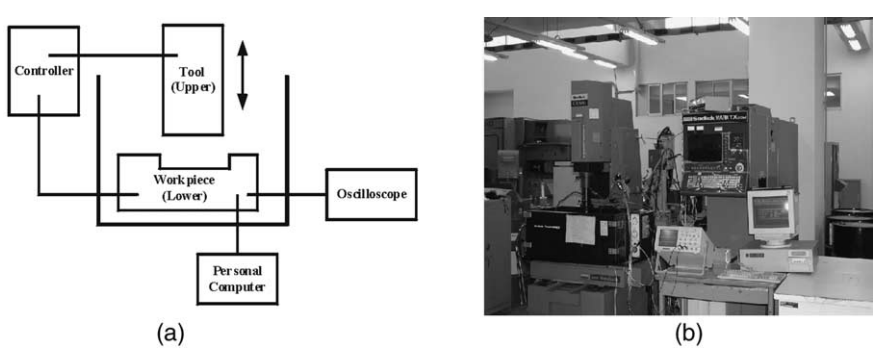

Fig. 7. Schematic drawing and photograph of the experimental equipment; (a) shows the schematic, and (b) shows the EDM machine and the personal computer.

of the tool and the works are tabulated in Table 2. In order to produce adequate data for model training, eighty sets of experimental conditions were arranged on the EDM machine. And, each set of the conditions was measured by running forty consecutive tests both on the case of copper–aluminum and copper–iron run.

Known for its capabilities on establishing the neural-networks models, MATLAB with associate

Table 1
Pertinent process parameters and values for the experiments

Symbol	Factor	Level				
		1	2	3	4	5
PL	Polarity of upper electrode	−	+			
ON	Discharge time (μs)	20	30	60	100	
Ip	Main power peak current (A)	12	22.5	30	39	48
An	Tool material (upper electrode)	Cu				
Ca	Work material (lower electrode)	Al	Fe			
OFF	Quiescent time (μs)	60				
V	Main power voltage (V)	90				
SV	Servo standard voltage	2				

Table 2
Physical characteristics and mechanical dimensions of the tool and the work

Materials	Composition	Density (kg/m^3)	Machined roughness R_{max} (μm)	Dimensions
Tool				
Cu	>99.95%	8896.6	4.02	ϕ9.5×50 mm
Work				
Fe	>99.9%	7870	1.08	□20×12 mm
Al	>99.5%	2,699	2.76	□20×12 mm

toolboxes, copyrighted by the MathWork Inc. in USA, was used for coding the algorithms. Also, with the help of Pentium III processors on a personal computer, the programs could be executed and finished in a few minutes.

4. Results and discussions

Before applying the neural networks for modeling the EDM process, we first need to decide the architecture and the topology of the networks; e.g. the number of hidden layers and the number of neurons in each layer in the networks. Based on the previous experiences from the work on semi-empirical model [27], five inputs and one output in the networks would be sufficient for this study. Therefore, the number of neurons in the input and output layer should be set to five and one, respectively. Also, the back-propagation architecture with one hidden layer is enough for the majority of applications, because it can form arbitrary mapping between a set of given inputs and outputs [2]. Therefore, one hidden layer for the neural networks was adopted. For determining the optimal number of neurons in hidden layer, a procedure was employed for optimizing the number of neurons in the hidden layer for the neural networks based on the results from

5000 epochs as shown in Fig. 8. By comparing the results, the number of hidden neurons was found to be 10 for the LOGMLP model, 12 for the TANMLP model, nine for the RBFN model, 20 for the Error TANMLP model, 50 for the Adaptive TANMLP model, and nine for the Adaptive RBFN model, respectively. It is noted that the TANMLP, RBFN and the Adaptive RBFN models have smaller check error than the others. And, the final results for the three networks versus

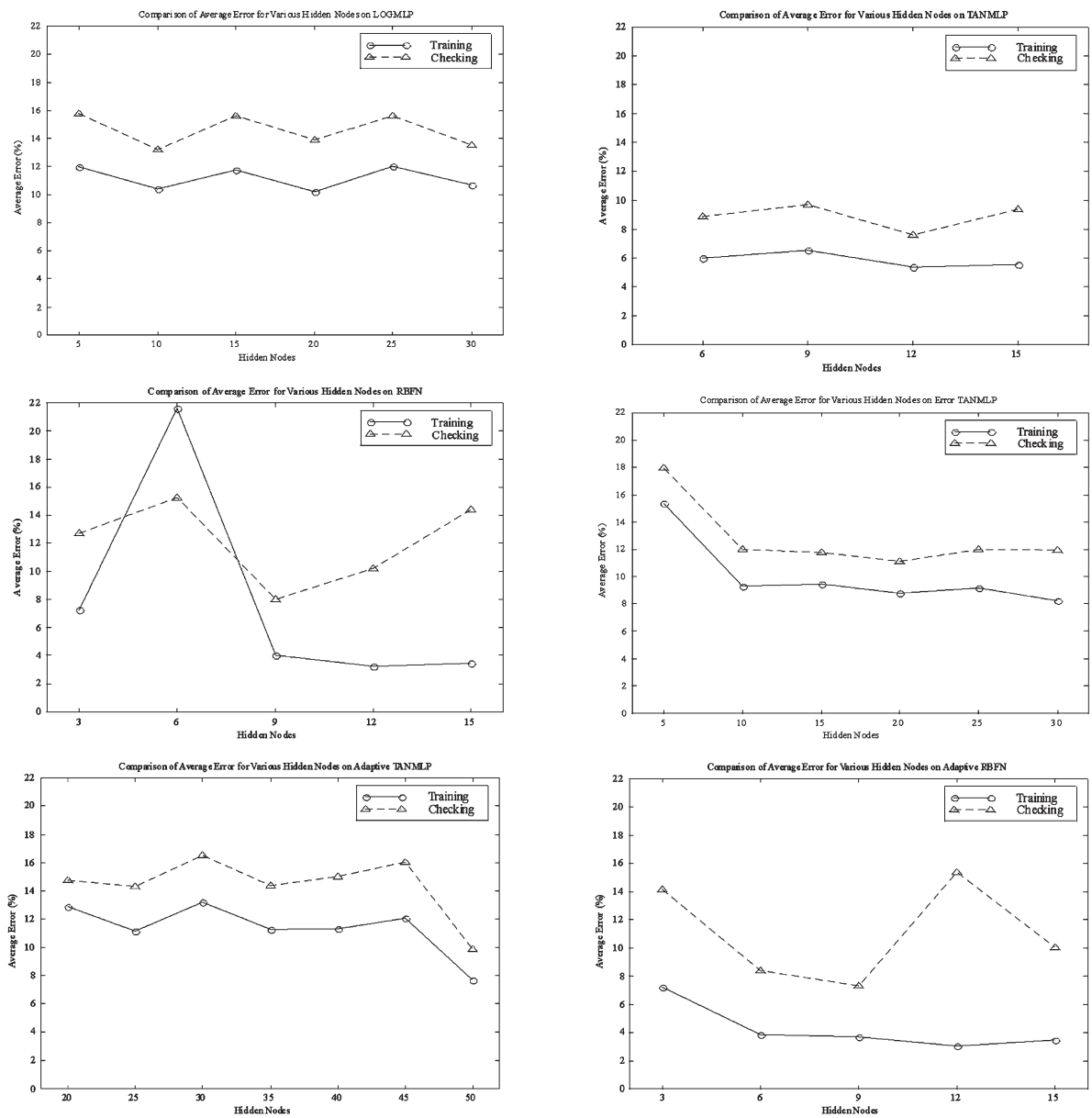

Fig. 8. Plots of comparison on average model error among the various models versus numbers of hidden nodes.

Table 3
Final comparison results on surface finish based upon the TANMLP model with the number of hidden nodes as the variable

Hidden nodes	6	9	12	15
Epoch in optimal	5000	5000	5000	5000
RMSE	0.03290	0.03699	0.03384	0.03393
Running time (s)	386	399	414	426
Average training error (%)	5.97	6.52	5.39	5.53
R^2	0.9632	0.9535	0.9611	0.9609
Average checking error (%)	8.87	9.70	7.60	9.35

Table 4
Same as Table 3 except for the RBFN

Hidden nodes	3	6	9	12
Epoch in optimal	5000	5000	5000	5000
RMSE	0.03183	0.01887	0.01809	0.01916
Running time (s)	199	223	250	276
Average training error (%)	7.45	4.38	4.01	4.21
R^2	0.9656	0.9879	0.9889	0.9875
Average checking error (%)	13.28	12.08	8.00	9.16

various numbers of hidden neurons are shown and compared in Tables 3–5. In these three tables, the prediction, the training and the check errors, are defined as follows,

$$\text{Error in } \% = \left| \frac{\text{Experimental Results} - \text{Predictions}}{\text{Experimental Results}} \right| \times 100(\%). \tag{7}$$

It should be noted that the check errors of the best cases are 7.60% for the TANMLP model, 8.00% for the RBFN model, and 7.29% for the Adaptive RBFN model, respectively.

Based on the MATLAB user's guide, there are eight different membership functions supported

Table 5
Same as Table 3 except for the adaptive RBFN

Hidden nodes	3	6	9	12
Epoch in optimal	5000	5000	5000	5000
RMSE	0.03814	0.02449	0.01725	0.01452
Running time (s)	199	224	248	275
Average training error (%)	9.40	3.56	3.68	3.29
R^2	0.9506	0.9796	0.9899	0.9928
Average checking error (%)	14.25	9.50	7.29	11.69

Table 6
Number of intrinsic parameters required for the ANFIS with various membership functions

Membership function	2 Bell MFs	2 Gaussian MFs	3 Bell MFs	3 Gaussian MFs
Number of inputs	5	5	5	5
Number of nodes	92	92	524	524
Number of linear parameters	192	192	1458	1458
Number of nonlinear parameters	30	20	45	30
Total number of parameters	222	212	1503	1488
Number of training data pairs	80	80	80	80
Number of checking data pairs	10	10	10	10
Number of fuzzy rules	32	32	243	243

in the MATLAB Fuzzy Logic toolbox [26]. In this paper, two and three of both the Bell and the Gaussian membership functions were employed. The characteristics of these types of membership functions are shown in Table 6. Evidently, the run-time of the computer will increase as the number of the membership function increases because number of the nodes, the parameters, and the if–then rules grows exponentially. From Tables 7–10, the final results of the ANFIS model versus various epochs corresponding to each of the cases are tabulated with bold-faced columns indicating the best one. They are one epoch for the two Bell MFs and the two Gaussian MFs, 300 epochs for the three Bell MFs, and five epochs for the three Gaussian MFs, respectively. As a result, the best model can be obtained after only one epoch when using two Bell MFs. Also, the convergence rate is higher than all the others. It is shown that advantage of quick convergence characteristic for ANFIS model has been observed when the supervised data are adequately consistent. This case was later used for predictions on the surface finish together with the three neural-network models (i.e. TANMLP, RBFN, and Adaptive RBFN) and the semi-empirical model [26] shown as follows:

Table 7
Final comparison results on the ANFIS model with two Bell-shape membership functions; the bold-faced column indicates the best case

Epochs	**1**	20	40	60	120	250	350	450
Final RMSE	**0.01312**	0.01152	0.00959	0.00891	0.00856	0.00850	0.00846	0.00843
Running time (s)	**1.16**	22	43	65	131	268	396	505
Average training error (%)	**2.51**	2.18	2.06	1.89	1.83	1.81	1.80	1.79
R^2	**0.9942**	0.9955	0.9969	0.9973	0.9975	0.9975	0.9976	0.9976
Average checking error (%)	**8.19**	12.58	25.72	20.78	17.29	17.48	16.88	17.22

Table 8
Same as in Table 7 except with two Gaussian membership functions

Epochs	**1**	20	40	60	120	250	350	450
Final RMSE	**0.01147**	0.00977	0.00922	0.00909	0.00890	0.00861	0.00859	0.00858
Running time (s)	**1.17**	22	44	64	128	271	386	495
Average training error (%)	**2.34**	2.05	1.97	1.93	1.88	1.84	1.83	1.83
R^2	**0.9955**	0.9968	0.9971	0.9972	0.9973	0.9975	0.9975	0.9975
Average checking error (%)	**11.85**	21.52	21.31	20.68	19.56	18.64	18.23	17.87

Table 9
Same as in Table 7 except with three Bell-shape membership functions

Epochs	1	20	80	100	200	**300**	350	450
Final RMSE	0.00983	0.00646	0.00216	0.00216	0.00205	**0.00204**	0.00203	0.00202
Running time (s)	119	2247	8966	11213	22358	**33461**	39038	50213
Average training error (%)	2.03	1.32	0.32	0.29	0.29	**0.29**	0.29	0.28
R^2	0.9967	0.9986	0.9999	0.9999	0.9999	**0.9999**	0.9999	0.9999
Average checking error (%)	36.03	23.93	11.52	11.28	11.28	**11.22**	11.25	11.30

Table 10
Same as in Table 7 except with three Gaussian membership functions

Epochs	1	**5**	11	20	120	250	350	450
Final RMSE	0.00852	**0.00805**	0.00727	0.00596	0.00394	0.00348	0.00346	0.00346
Running time (s)	132	**575**	1264	2293	13717	28553	39901	51343
Average training error (%)	1.78	**1.70**	1.55	1.33	0.90	0.73	0.72	0.72
R^2	0.9975	**0.9978**	0.9982	0.9988	0.9995	0.9996	0.9996	0.9996
Average checking error (%)	18.85	**18.07**	19.39	18.90	20.71	24.37	25.75	25.83

$$R_a = A_1\left[\frac{\alpha}{H_{v^{1/2}}}\right]\left(\frac{I_p}{\alpha^{1/2}\rho^{1/2}\alpha^{3/2}}\right)^{a_1}\left(\frac{\tau_{on}H_v}{\alpha}\right)^{b_1}\left(\frac{E}{\rho\alpha^2 H_{v^{1/2}}}\right)^{c_1}(J_a)^{d_1} \tag{8}$$

where R_a is the surface finish of work, A_1 being a constant dependent of work materials, $\alpha=\kappa/\rho C_p$ being the thermal diffusivity, and H_v being the latent heat of evaporation. In the thermal diffusivity

of the work materials, κ is the thermal conductivity and C_p is the specific heat capacity with ρ being the density. As for the electrical energy, E is the input energy to work and I_p is the peak current together with τ_{on} being the discharge time; while σ is the electric conductivity of the work. Being defined by $J_a=T_vC_p/H_v$, J_a is the Jacob number of the work materials with T_v being the vaporization temperature. This semi-empirical model has been established by employing dimensional analysis based upon pertinent process parameters screened by the Design of Experiment Method [26].

Together with all the models, Table 11 shows the final results for modeling the surface finish in EDM process. It is noted that the best one is the Adaptive RBFN model. Compared to the TANMLP model and RBFN model, Adaptive RBFN model has faster convergence and better performance based on the 5000 epochs case as shown in Fig. 9. For detail illustrations, the final parametric results of the Adaptive RBFN model are shown in Figs. 10 and 11. In these figures, the minimum average check error has been 7.29%. In Fig. 11, the training differences are less than 0.5 μm for surface finish on the work.

As a further step for verifications, experiments were scheduled with process parameters set to the border on the training process window. The comparisons among all the models are shown in Table 12. In particular, the semi-empirical model, the TANMLP model, the RBFN model, the adaptive RBFN model, and the ANFIS model with two Bell membership functions are plotted

Table 11
Final comparison results on surface finish based upon various models; Cu–Al, Fe are employed for the tool and work materials in the table, respectively

	LOGMLP	TANMLP	RBFN	ERROR TANMLP	Adaptive TANMLP
Hidden nodes	10	12	9	20	50
Epochs	5000	5000	5000	5000	5000
Final RMSE	0.04840	0.03384	0.01809	0.05812	0.04472
Running time (s)	363	414	250	462	599
Average training error (%)	10.4274	5.394	4.01	8.8153	7.6571
R^2	0.9204	0.9611	0.9889	0.8853	0.9321
Average checking error (%)	13.19	7.60	8.00	11.12	9.86
	Adaptive RBFN	ANFIS (2 Bell MF)	ANFIS (2 Gaussian)	ANFIS (3 Bell MF)	ANFIS (3 Gaussian)
Hidden nodes	9				
Epochs	5000	1	1	300	5
Final RMSE	0.01725	0.01312	0.01147	0.00204	0.00805
Running time (s)	248	1.19	1.17	33461	575
Average training error (%)	3.68	2.51	2.34	0.29	1.70
R^2	0.9899	0.9942	0.9955	0.9999	0.9978
Average checking error (%)	7.29	8.19	11.85	11.22	18.07

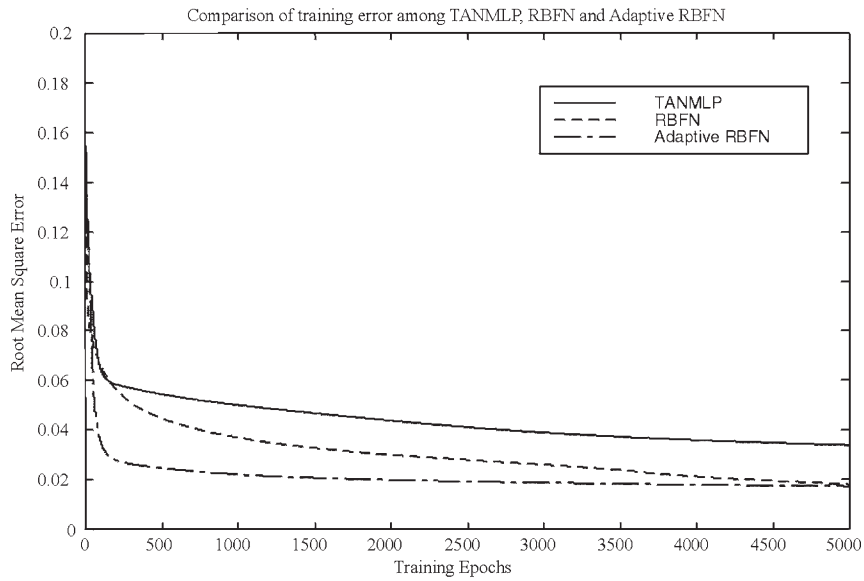

Fig. 9. Plots of comparison on training errors between the TANMLP, the RBFN, and the Adaptive RBFN; the Adaptive RBFN model is better than the other models.

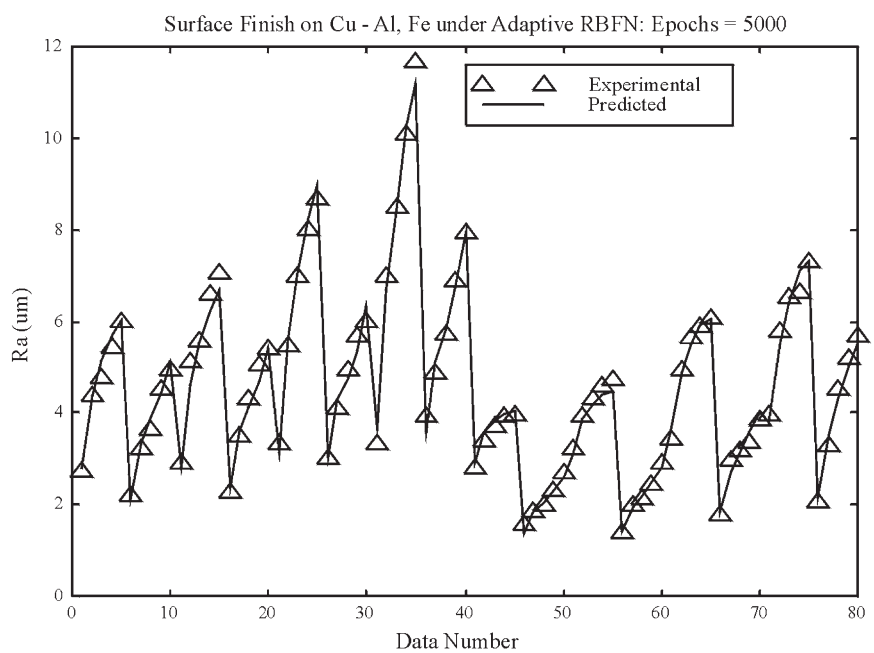

Fig. 10. Plots of comparison results between the measured and predicted surface finish based upon the Adaptive RBFN model.

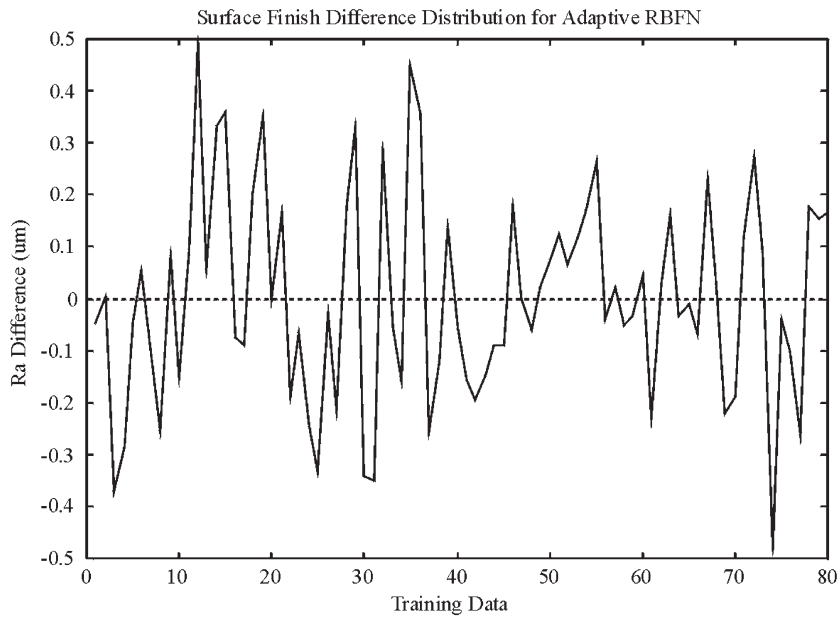

Fig. 11. Plots of differences in surface finish calculated based upon Fig. 10 versus the number of training data after completing the training procedure.

on the same scale as shown in Fig. 12. Although the TANMLP model is the best one with 8.58% check error both shown in Table 12 and Fig. 12, the other models have the same percentage of check error except for the semi-empirical model. That means the predictions of surface finish in EDM process by making use of the TANMLP model, the RBFN model, the Adaptive RBFN model, and the ANFIS model have all shown in good agreement with the experimental results; but, the convergence rate of the ANFIS method is obviously better than the other methods.

5. Conclusions

In this paper, seven models for predictions of surface finish of work in EDM process have been established and compared based upon six neural networks and a neuro-fuzzy network with pertinent machine process parameters given by the DOE method. The networks, namely the LOGMLP, the TANMLP, the RBFN, the Error TANMLP, the Adaptive TANMLP, the Adaptive RBFN, and the ANFIS have been trained and compared by the same experimental data together with the change of electrode polarity condition. According to the comparisons on the training results, it has been shown that the TANMLP, the RBFN, the Adaptive RBFN, and the ANFIS models are more accurate than the other models. Also, further experimental verifications have shown that the predictions on the surface finish have gone down to 8.58% average error on the TANMLP model. Also, the other models such as the RBFN, the Adaptive RBFN, and the ANFIS have the same magnitude of average error. Conclusively speaking, the surface finish of work in the EDM process can be predicted by the above models with reasonable accuracy.

Based on authors' previous experiences with semi-empirical models, it is comfortable to con-

Table 12
Experimental verifications based upon different process parameters indicated in the note below the table used for training procedure for the three different models

Case	Materials	Experiment	Semi-empirical model	Error (%)	TANMLP	Error (%)	RBFN	Error (%)	Adaptive RBFN	Error (%)	ANFIS (2 Bell MF)	Error (%)
1	Cu(+):Al(−)[a]	3.3375	4.2165	26.34	3.6210	8.49	3.1880	4.48	3.1339	6.10	3.3104	0.81
2	Cu(+):Al(−)[b]	5.9675	7.6650	28.45	7.2836	22.06	6.9555	16.56	6.7573	13.23	7.1401	19.65
3	Cu(−):Al(+)[b]	5.5300	5.2054	5.87	5.6576	2.31	5.4029	2.30	5.6514	2.19	5.7149	3.34
4	Cu(+):Al(−)[c]	8.5375	9.0243	5.70	8.3893	1.74	8.7082	2.00	8.5997	0.73	9.5223	11.54
5	Cu(−):Al(+)[c]	6.7075	6.1285	8.63	6.7437	0.54	6.3178	5.81	6.6696	0.56	6.1722	7.98
6	Cu(−):Fe(+)[d]	2.2025	2.9172	32.45	2.3337	5.96	2.2200	0.79	2.1778	1.12	2.1525	2.27
7	Cu(+):Fe(−)[b]	6.4675	5.1171	20.88	5.8979	8.81	5.3930	16.61	5.1451	20.45	6.1067	5.58
8	Cu(−):Fe(+)[b]	3.4675	4.0119	15.70	3.6016	3.87	3.6112	4.14	3.4677	0.01	3.6758	6.01
9	Cu(+):Fe(−)[c]	7.3325	6.0246	17.84	6.8428	6.68	6.3285	13.69	6.0817	17.06	7.0048	4.47
10	Cu(−):Fe(+)[c]	3.8900	4.7234	21.42	4.4967	15.60	4.4208	13.64	4.3363	11.47	4.6792	20.29
11	Ag(+):Al(−)[b]	6.6350	7.5472	13.75	7.2836	9.78	6.9555	4.83	6.7573	1.84	6.9351	4.52
12	Ag(−):Al(+)[b]	5.8550	5.1254	12.46	5.6576	3.37	5.4029	7.72	5.6514	3.48	5.6688	3.18
13	Ag(+):Fe(−)[b]	6.2650	5.0385	19.58	5.8979	5.86	5.3930	13.92	5.1451	17.87	6.0238	3.85
14	Ag(−):Fe(+)[b]	4.1450	3.9503	4.70	3.6016	13.11	3.6112	12.88	3.4677	16.34	3.5906	13.38
15	Ag(−):Al(+)[e]	7.7125	6.0343	21.76	6.7437	12.56	6.3178	18.08	6.6696	13.52	6.1117	20.76
16	Ag(+):Al(−)[f]	6.3300	6.9830	10.32	6.7596	6.79	6.8680	8.50	7.0266	11.01	6.7726	6.99
17	Cu(+):Fe(−)[g]	7.1500	6.0246	15.74	6.8428	4.30	6.3285	11.49	6.0817	14.94	7.0048	2.03
18	Ag(−):Fe(+)[h]	3.7350	4.7835	28.07	4.5807	22.64	4.3450	16.33	4.4683	19.63	4.7001	25.84
Average error (%)				17.20		8.58		9.65		9.53		9.03

[a] τ_{on}=60 μs; Ip=12 A; τ_{off}=120 μs; *V*=90 V; SV=2.
[b] τ_{on}=120 μs; Ip=22.5 A; τ_{off}=60 μs; *V*=120 V; SV=2.
[c] τ_{on}=120 μs; Ip=30 A; τ_{off}=120 μs; *V*=120 V; SV=2.
[d] τ_{on}=60 μs; Ip=30 A; τ_{off}=60 μs; *V*=120 V; SV=2.
[e] τ_{on}=120 μs; Ip=30 A; τ_{off}=30 μs; *V*=90 V; SV=2.
[f] τ_{on}=60 μs; Ip=30 A; τ_{off}=60 μs; *V*=120 V; SV=2.
[g] τ_{on}=120 μs; Ip=30 A; τ_{off}=120 μs; *V*=60 V; SV=2.
[h] τ_{on}=60 μs; Ip=48 A; τ_{off}=30 μs; *V*=60 V; SV=2.

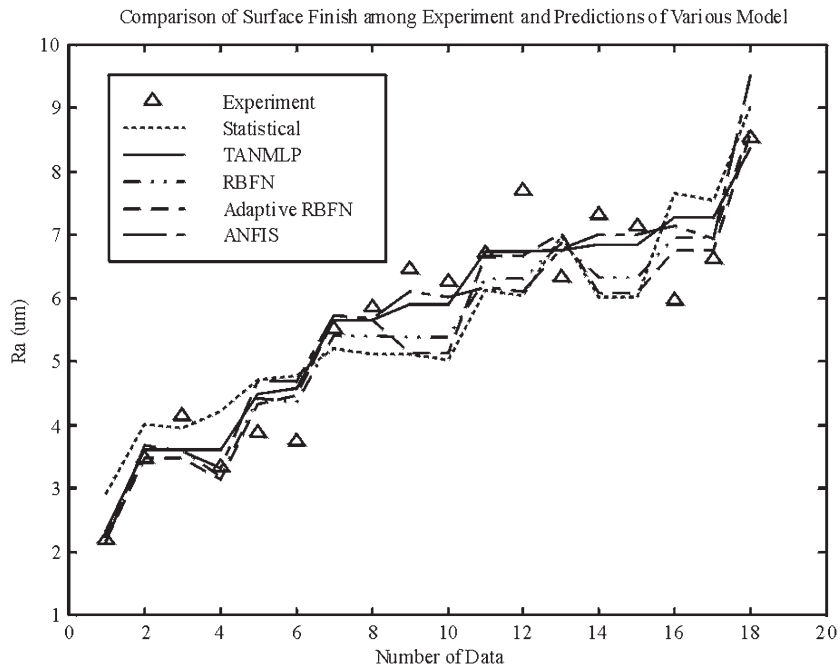

Fig. 12. Plots of comparison on surface finish among the measured data and predictions based upon various models; the TANMLP model shows better predictions than the other models.

clude that the surface finish of work in EDM process can be successfully modeled based on the neural networks even though the EDM process has been known for its stochastic nature. This paper has successfully established some new process models, which consist of the pertinent process parameters, for predicting the surface finish of work. For practical applications, exact machining time and values of process parameters could be better controlled on the shop floor if the process models are employed in the industry.

References

[1] J.A. Freeman, D.M. Skapura, Neural networks: algorithms, applications, and programming techniques, Addison-Wesley, Reading. MA, 1992.
[2] L. Fausett, Fundamentals of neural networks: architectures, algorithms, and applications, Prentice-Hall, Englewood Cliffs, NJ, 1994.
[3] S. Rangwala, D. Dornfeld, Sensor integration using neural networks for intelligent tool condition monitoring, J. Eng. Ind., ASME 112 (1990) 219–228.
[4] O. Masory, Monitoring machining processes using multi-sensor reading fused by artificial neural network, J. Mater. Proc. Tech. 28 (1991) 231–240.
[5] I.N. Tansel, C. Mekdeci, O. Rodriguez, B. Uragun, Monitoring drill conditions with wavelet based encoding and neural networks, Int. J. Mach. Tools Manufact. 33 (1993) 559–575.
[6] Y.S. Tarng, Y.W. Hseih, S.T. Hwang, An intelligent sensor for monitoring milling cutter breakage, Int. J. Adv. Manuf. Tech. 9 (1994) 141–146.

[7] Y.S. Tarng, Y.W. Hseih, S.T. Hwang, Sensing tool breakage in face milling with a neural network, Int. J. Mach. Tools Manufact. 34 (1994) 341–350.
[8] I.N. Tansel, A. Wagiman, A. Tziranis, Recognition of chatter with neural network, Int. J. Mach. Tools Manufact. 31 (1991) 539–552.
[9] Y.S. Tarng, T.C. Li, M.C. Chen, On-line drilling chatter recognition and avoidance using an ART2-A neural network, Int. J. Mach. Tools Manufact. 34 (1994) 949–957.
[10] B.Y. Lee, Y.S. Tarng, S.C. Ma, Modeling of the process damping force in chatter vibration, Int. J. Mach. Tools Manufact. 35 (1995) 951–962.
[11] X.Q. Li, Y.S. Wong, A.Y.C. Nee, A comprehensive identification of tool failure and chatter using a parallel multi-ART2 neural network, J. Manufact. Sci. Eng., ASME 120 (1998) 433–442.
[12] V. Cariapa, K.S. Akbay, R. Rudraraju, Application of neural networks for compliant tool polishing operations, J. Mater. Proc. Tech. 28 (1991) 241–250.
[13] Y.S. Tarng, S.T. Hwang, Y.S. Wang, A neural network controller for constant turning force, Int. J. Mach. Tools Manufact. 34 (1994) 453–460.
[14] Y.S. Tarng, S.C. Ma, L.K. Chung, Determination of optimal cutting parameters in wire electrical discharge machining, Int. J. Mach. Tools Manufact. 35 (1995) 1693–1701.
[15] T.W. Liao, L.J. Chin, A neural network approach for grinding process: modeling and optimization, Int. J. Mach. Tools Manufact. 34 (1994) 919–937.
[16] B.Y. Lee, H.S. Liu, Y.S. Tarng, Modeling and optimization of drilling process, J. Mater. Proc. Tech. 74 (1998) 149–157.
[17] T.J. Ko, D.W. Cho, Adaptive optimization of face milling operations using neural networks, J. Manufact. Sci. Eng., ASME 120 (1998) 443–451.
[18] J.Y. Kao, Y.S. Tarng, A neural network approach for the on-line monitoring of the electrical discharge machining process, J. Mater. Proc. Tech. 69 (1997) 112–119.
[19] H.S. Liu, Y.S. Tarng, Monitoring of the electrical discharge machining process by adaptive networks, Int. J. Adv. Manufact. Tech. 13 (1997) 264–270.
[20] Z. Katz, J. Naude, A neural network/expert system approach for design improvement of products manufactured by EDM, J. Manufact. Sci. Eng., ASME 121 (1999) 733–738.
[21] K.M. Tsai, P.J. Wang, Study on parameters optimization for electric discharge machining, The 14th Nat. Conf. on Mech. Eng., vol. 5, Chin. Soc. Mech., Toayuan, Taiwan, 1997, p. 165–71.
[22] J.S.R. Jang, C.T. Sun, E. Mizutani, Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence, Prentice-Hall, Englewood Cliffs, NJ, 1997.
[23] N.B. Karayiannis, A.N. Venetsanopoulos, Artificial neural networks: learning algorithms, performance evaluation, and applications, Kluwer Academic, Amsterdam, 1993.
[24] J.M. Zurada, Introduction to artificial neural systems, West, 1992.
[25] J.S.R. Jang, ANFIS: Adaptive-Network-Based Fuzzy Inference System, IEEE Trans. Syst. Man Cyb. 23 (1993) 665–685.
[26] N. Gulley, J.S.R. Jang, Fuzzy logic toolbox: for use with MATLAB—user's guide, The MathWorks, USA, 1995.
[27] P.J. Wang, K.M. Tsai, Semi-empirical model of surface finish on electrical discharge machining, Int J Mach Tools Manufact (in press).