EXPERT SOFTWARE FOR IMPROVING NONCONVENTIONAL PROCESSING PARAMETERS

Tiberiu Marius KARNYANSZKY[1], Dan Laurențiu LACRĂMĂ[2]

ABSTRACT

This paper is focused on the improving of the nonconventional processing parameters using computers and expert software. All over the world the nonconventional processing is used in cases where traditional techniques is too complex or too expensive, because the steel is very hard. In such situations non conventional methods like electro erosion, electrochemical erosion, complex electrochemical erosion and laser erosion could be the solution.

KEYWORDS: complex electrochemical erosion, neural networks

1. THEORETICAL CONSIDERATIONS

In order to develop a program that automatically performs the functions' settling of the dependence of the technological parameters on the influencing factors, we have considered the following mathematical patterns with polynomial functions. Concretely, let us consider the dependences as being of one (1, 2, 3) and two variables (4, 5) only, namely:

$$z=a_0+a_1\cdot x \quad (1)$$
$$z=a_0+a_1\cdot x+a_2\cdot x^2 \quad (2)$$
$$z=a_0+a_1\cdot x+a_2\cdot x^2+a_3\cdot x^3 \quad (3)$$
$$z=a_0+a_1\cdot x+a_2\cdot y \quad (4)$$
$$z=a_0+a_1\cdot x+a_2\cdot y+a_3\cdot x^2+a_4\cdot y^2+a_5\cdot x\cdot y \quad (5)$$

the establishing of the coefficients a_0, a_1, …being based on the smallest squares method ([1]).

2. OBTAINING THE PATTERN USING MATHEMATICAL METHODS

We have obtained the following mathematical patterns of the dependence of the EEC processing productivity (Q_p) on the current density (j) and on the relative speed between PO and TO (v_r), at the debiting of the metallic carbures using OT of OL ([2], using THE experimental data presented in Table 1):

- P10 debiting (figure 1, only the dependency between Q_p and j with v_r=6):

$$Q_p=0{,}06145-0{,}9006\cdot j+9{,}55098\cdot j^2-18{,}45541\cdot j^3 \text{ (error 4\%)} \quad (6)$$

- P10 debiting (figure 2, only the dependency between Q_p and v_r with j=0.08):

$$Q_p=-0{,}06929-0{,}00872\cdot v_r+83\cdot 10^{-5}\cdot v_r^2-2\cdot 10^{-5}\cdot v_r^3 \text{ (error 5\%)} \quad (7)$$

- P10 debiting (figure 3):

$$Q_p=-0{,}1376+1{,}3513\cdot j+0{,}0136\cdot v_r-2{,}155\cdot j^2-0{,}0004\cdot v_r^2-0{,}0055\cdot j\cdot v_r \text{ (error 16,58\%)} \quad (8)$$

- P20 debiting (figure 4):

$$Q_p=-0{,}0674+1{,}2659\cdot j+0{,}0058\cdot v_r-2{,}6495\cdot j^2-0{,}0002\cdot v_r^2+0{,}0130\cdot j\cdot v_r \text{ (error 9,33\%)} \quad (9)$$

- P30 debiting (figure 5):

$$Q_p=-0{,}1168+0{,}6536\cdot j+0{,}0133\cdot v_r+1{,}2880\cdot j^2-0{,}0003\cdot v_r^2+0{,}005\cdot j\cdot v_r \text{ (error 22.21\%)} \quad (10)$$

- P40 debiting (figure 6):

$$Q_p=-0{,}0303+0{,}2857\cdot j+0{,}0092\cdot v_r+0{,}24\cdot j^2-0{,}0003\cdot v_r^2+0{,}032\cdot j\cdot v_r \text{ (error 12.42\%)} \quad (11)$$

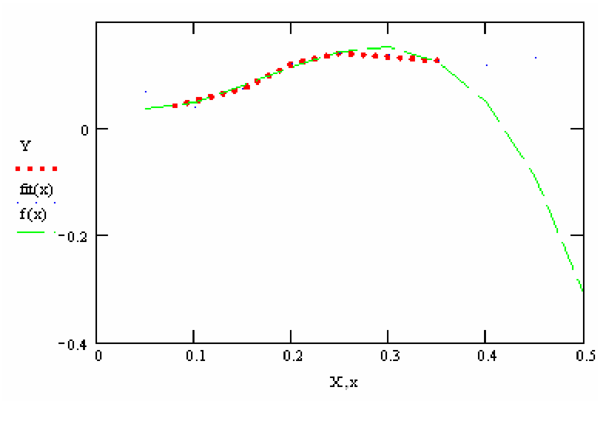


Fig. 1. P10 debiting, Q_p dependency on j

Table 1. P10 debiting results

j	v_r	Q_p
0.08	6	0.0418
	10	0.0491
	15	0.0567
	20	0.0752
	27	0.0592
0.15	6	0.0750
	10	0.0930
	15	0.1125
	20	0.1357
	27	0.0994
0.20	6	0.1219
	10	0.1132
	15	0.1212
	20	0.1712
	27	0.1011
0.25	6	0.1416
	10	0.1573
	15	0.1805
	20	0.2063
	27	0.1069
0.35	6	0.1253
	10	0.1312
	15	0.1632
	20	0.1753
	27	0.1156

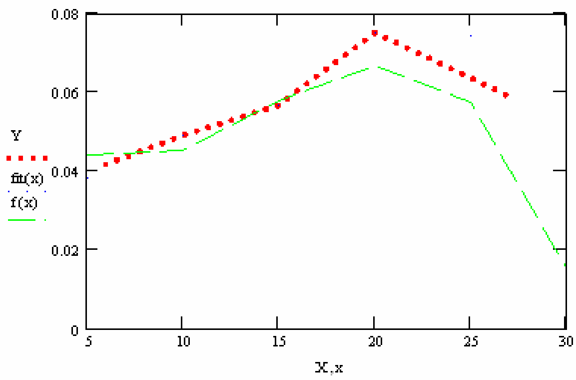


Fig. 2. P10 debiting, Q_p dependency on v_r

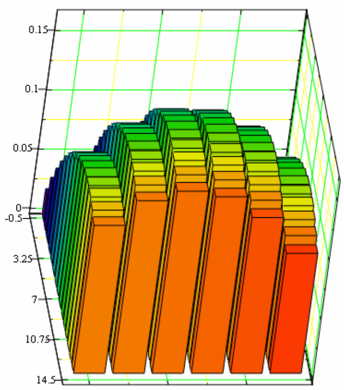


Fig. 3. P10 debiting, Q_p dependency on j and v_r

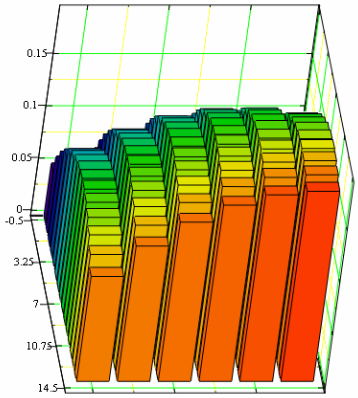


Fig. 4. P20 debiting, Q_p dependency on j and v_r

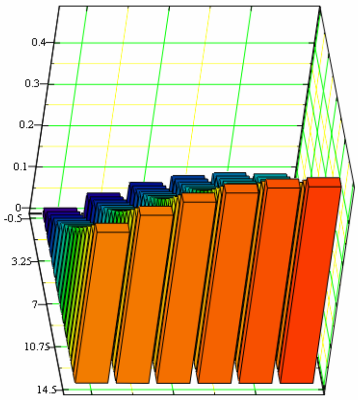


Fig. 5. P30 debiting, Q_p dependency on j and v_r

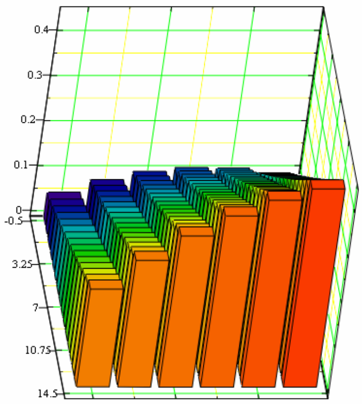


Fig. 6. P30 debiting, Q_p dependency on j and v_r

3. OBTAINING THE PATTERN USING NEURAL NETWORKS

In order to solve the above-mentioned task, the authors of this paper selected the Multilayer Perceptron trained with the error back propagation rule. As stated in the scientific literature MLP is a simple and powerful tool that can be applied successfully to solve many problems.

The *Universal approximation theorem* formulated by Cybenko proved rigorously that a single hidden layer MLP is sufficient to uniformly approximate any continuous function with support in a unit hypercube. Thus, a single hidden layer neural net should be good enough to obtain a satisfactory solution to the CEE parameters correlation problem.

Nevertheless the Universal approximation theorem has a limited practical value. The neurons inside the unique hidden layer tend to interact with each other globally. Therefore in complex situations this interaction makes it difficult to improve the approximation at a certain point without worsening it at some others. In practice, two or more hidden layers can prove useful in order to make the approximation process into a more manageable one.

Consequently the authors decided to experiment and compare the results of three alternative neural architectures:

a. Single hidden layer MLP;
b. Two hidden layers MLP;
c. Three hidden layers MLP.

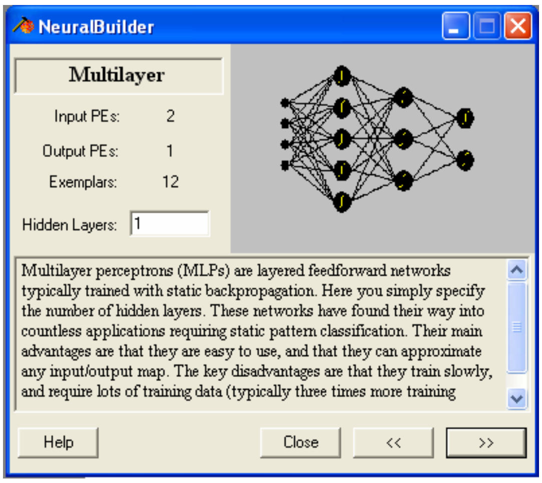


Fig. 7. Neural Builder GUI window, Neuro Solutions 4.31

All the three neural nets were implemented using the Neuro Solutions 4.31 software from Neuro Dimensions Inc. This integrated environment provided us the possibility to quickly build, train and test the networks using a simple and efficient set of GUI and results windows as shown in Figure 2.1. Each of the three neural nets architecture is depicted in Figure 2.2 as shown in the Neuro Solutions user screen.

As stated before each of the three neural nets is able to assure a reasonably good approximation of the curve t_p=f(I,W_{TO}), but the authors tried to find out which of them is the best solution both in terms of precision and efficiency.

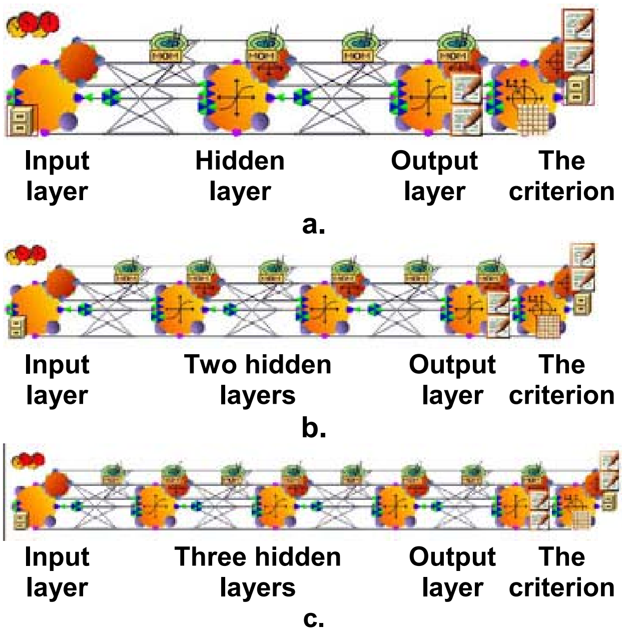
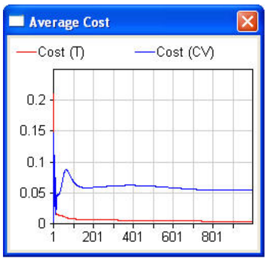


Fig. 8. Neuro nets architecture

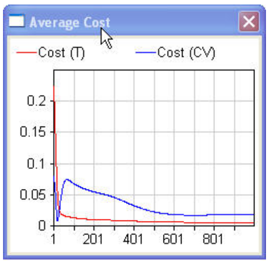
Using the three structures represented in Fig. 8 the authors performed the experiments with the same training set containing 388 data samples. Each sample consists of the directly measured technological parameters of the debiting process on a real CEE machine tool (i.e. t_p, I and W_{TO}). All data were collected from the same equipment using only OL37 stainless steel samples. The data collecting procedure was made in accordance with the rules stated in Neural Networks usage. The CV and T average costs together with the number of minimum necessary training epochs are shown for all the three neural nets in Figure 9. The test set was the same for all the three nets and it was performed with 82 different data samples.

Table 2. Neural results

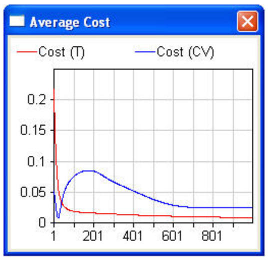
Neural net Structure	Incorrect estimations	
	Samples	[%]
One hidden layer	5	93.902
Two hidden layer	3	96,342
Three hidden layer	2	97.561



Single hidden layer MLP:
CV Avg. Cost ≅ 0.05
Training epochs ≅ 760



Two hidden layers MLP:
CV Avg. Cost ≅ 0.02
Training epochs ≅ 580



Three hidden layers MLP:
CV Avg. Cost ≅ 0.02
Training epochs ≅ 670

Fig. 9. CV and T Average Costs

4. CONCLUSIONS

By analyzing the determined functions there can be observed that:

- the Q_p dependence on j (only) using the 3 rank functions is correct with an maximum 4% error;
- the Q_p dependence on v_r (only) using the 3 rank functions is correct with an maximum 5% error;
- the Q_p dependence on j and r_s (relative speed) using the 2 rank functions is correct with an maximum 22% error;
- Q_p can be expressed both depending on j and r_s;
- Q_p depends more on j than on r_s, both due to the 1 rank component and to the 2 rank one;
- j can be used to control Q_p better than r_s.

The most important concluding comment of the above results is that the use of neural nets produces a significant improvement from the method of curve fitting with the third rank polynomial functions. This progress was achieved without employing great programming effort or extensive time-consuming computations. Analyzing the incorrect estimations in all the three cases some concluding remarks are obvious:

- The two hidden layers MLP is the best solution because it gives more precise results than the single hidden layer structure;
- The three hidden layers structure performs a little better at the testing stage, but the improvement is not significant and consequently the added costs are not worthwhile;
- Better results should be obtained with an enlarged number of data samples in the training set, but the data collection procedure involves a great effort and it is time consuming. Further improvements in performance could result from using a more flexible structure as RBF neural nets. This could lead to the development of a neural network able to solve the parameters control for a set of similar but different stainless steel qualities. The use of neural networks to manage mechanical processes parameters is a very useful practice, but finding the optimal solution is not straightforward and need a carefully work from the data collection stage to the final implementation, training and testing.

REFERENCES

[1] Tiberiu Marius Karnyanszky, *Contribuții la conducerea automată a prelucrării dimensionale prin eroziune electrică complexă*, Teză de doctorat, Universitatea "Politehnica" Timişoara, 2004
[2] Ştefan Kilyeni, *Metode numerice*, volumul I+II, Editura Orizonturi Universitare, Timişoara, 1997
[3] Zenoviu Lăncrăngean, *Contribuții la prelucrarea corpurilor de revoluție prin eroziune electrică complexă*, Teză de doctorat, Institutul Politehnic „Traian Vuia" Timişoara, 1986

AUTHORS

1. Assoc.Prof. Tiberiu Marius KARNYANSZKY, PhD., Dipl.Eng., "Tibiscus" University of Timisoara, Romania, mtk@tibiscus.ro, +40/744/599190
2. Assoc.Prof. Dan Laurențiu LACRĂMĂ, PhD., Dipl.Eng., "Tibiscus" University of Timisoara, Romania, dlacrama@tibiscus.ro,