4th International Conference on Mechanical Engineering,December 26-28, 2001, Dhaka,Bangladesh/pp.VI 61-66

CUTTING FORCE EVALUATION IN ORTHOGONAL MACHINING OF GLASS/EPOXY COMPOSITES – A NEURAL NETWORK ANALYSIS

D.Abdul Budan* A.Muruganandam and S.Vijayarangan*****
Dept of Mech Engg. PSG College of Technology.
COIMBATORE – 641 004 (TN), INDIA.

Abstract Cutting force is one among the basic factors, which measure the machinability of any material and decide on power required for machining. Experimental techniques are more commonly used to measure the cutting forces developed during machining. The present work is an attempt being made to save the time and expenses involved in experimentation. A Multi-layer Perceptron Feed Forward neural network is constructed to evaluate the cutting forces developed during the machining of Glass/ Epoxy composite material. There are many parameters, which have an effect on cutting forces. In view of minimizing the input variables, only three parameters having predominant effect were considered in this work. A finite element model was developed to evaluate the cutting forces and train the network. The neural network output results have shown good agreement with the results obtained from the modified Merchant's equation. The neural network outputs were compared with the desired output values; it was observed that maximum error reduction is possible.

INTRODUCTION

Fibre reinforced plastic (FRP) composite materials possess high strength and stiffness to weight ratio than the common structural materials and are extensively used in automobile, aircrafts, space vehicles, marine and general engineering applications. Despite the recent developments in the near net shape manufacturing, composite parts often require post-mold machining to meet dimensional tolerance, surface quality and other functional requirements. Cutting force is one among the basic factors, which measure the machinability of any material and decide on power required for machining. The determination of cutting forces by experimental means is more commonly adopted. However conducting an experiment to measure these forces every time is highly expensive and time consuming. Also in some cases, particularly in machining composites, preparing specimens with the required specifications is extremely difficult. To circumvent these difficulties, an effective orthogonal cutting model is needed to determine the cutting forces developed during machining of FRP materials.

In this work a Multi-layer Perceptron Feed Forward neural network is constructed to evaluate the cutting forces developed during the machining of Glass/ Epoxy composite material. The fibre orientation, fibre percentage and depth of cut were chosen as the input parameter for this purpose. There are many parameters, which have an effect on cutting forces. In view of minimizing the input variables, only three parameters having predominant effect were considered in this work. An FEA model was developed to evaluate the cutting force and feed force and to train the neural network.

Many of the literatures revealed that the selection of parameters for artificial neural network (ANN) is a major problem. Dornfeld et.al [2] reported that the parameters of the network have to be carefully selected for the precise output of the network. Dimla et al [3] have experimented different Perception networks with 5, 10, 20 hidden nodes by keeping the learning rate parameters and momentum factors as constant and achieved good success rate with more number of hidden nodes. However this has been contradicted by Yao and Fang [4]. They revealed that it is misleading to say that more hidden nodes will improve the performance and generally the number of hidden nodes is a critical and complicated factor.

In the present work 16 training patterns are considered. Suitable scale factor is chosen for the input variables to train the network. After attaining convergence, the trained weights are fed into the testing network model which is similar to that of training network except having only the capacity to determine the outputs for the corresponding input variables. It is observed that the output results of the neural network have shown good agreement with the modified Merchants model. Further the neural network outputs were compared with the desired output values, the errors were found to be almost converging.

Email: Abdul_budan@rediffmail.com* murug_a@lycos.com**
svrangan@mech.psgtech.ac.in***

MULTILAYER PERCEPTRON

The general architecture of a 3-layerd Perceptron is shown in fig.1. Multi-layered Perceptron (MLP) uses a Back-Propagation Algorithm (BPA) for training the network in a supervised manner. BPA is a steepest-decent method, where weight values are adjusted in an iterative fashion while moving along the error surface to arrive at minimal range of error when input patterns presented to the network for the learning. The learning process consists of two passes through different layers of the network, a forward pass and a backward pass. In the forward pass, the input pattern is applied to the nodes of the input layer and its effect propagates through the network, layer by layer. During the forward pass, synaptic weights are all fixed. The error, which is the difference between the actual output of the network and the desired output, is propagated back through backward pass to update the synaptic weights. The weights are continuously updated every time when the input patterns are presented to the network, the process continues till the actual output of the network comes closer to the desired output. If all the input patterns are propagated once through the network, it is called as cycle or epoch.

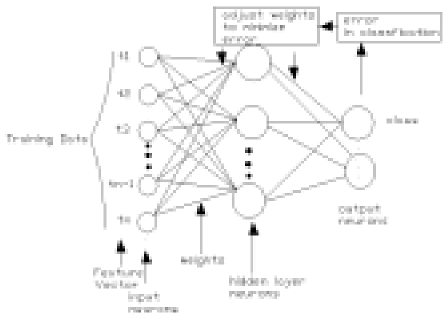


Fig.1. Multi-layered Perceptron

Steps involved in feed forward neural network technique [5]

Step 1: Decide the number of hidden layers.

Step 2: Decide the number of neurons for the input layer and for the output layer. For input layer the number of neurons is equal to the number of input variables and for output layer it is equal to the number of outputs required. Set small number of neurons for the hidden layer.

Step 3: Get the training input pattern.

Step 4: Assign small weight values for the neurons connected in between the input, hidden and output layers.

Step 5: Calculate the output for all the neurons in hidden and output layers using the following formula.

$$Out_i = f(net_i) = f\,(\,\Sigma\, w_{ij}\, Out_j + \theta_i).$$

Here, Out_i is the output of the i^{th} neuron in the layer under consideration, out_j is the output of the j^{th} neuron in the preceding layer. f is the sigmoid function that can be expressed as,

$f(net_I) = 1/\,(\,1+ e^{-neti\,/\,q}\,)$. q is termed as temperature.

Step 6: Determine the outputs at the output layer and compare these results with the desired output values to evaluate the error of the output neurons,

Error = desired output- actual output.

Similarly determine the root mean square error value of the output neurons.

$$E_p = ½\ \Sigma\,(\,t_{pj} - o_{pj}\,)^2.$$

Where E_p is the error for the p^{th} presentation vector, t_{pj} is the desired value for the j^{th} output neuron (i.e. the training set value) and o_{pj} is the desired output of the j^{th} output neuron.

Step 7: Determine the error available at the neurons of the hidden layer and back -propagate these errors to the weight values connected in between the neurons of hidden layer and input layer. Similarly, back-propagate the errors available at the output neurons to the weight values connected in between the neurons of hidden layer and output layer using the following formula.

Error $\delta_{pj} = (\,t_{pj} - O_{pj}\,)\, O_{pj}\,(1- O_{pj})$ for output neurons.
Error $\delta_{pj} = (\,t_{pj} - O_{pj}\,)\, O_{pj}\, \Sigma\, \delta_{pk}\, w_{kj}$ for hidden neurons.

Weight adjustment is made as follows,

$$\Delta\, w_{ji}\,(n+1) = \eta\,(\delta_{pj}\, O_{pi}) + \alpha\Delta w_{ji}\,(n).$$

η- Learning rate parameter.
α- Momentum.

Step 8: Go to step 3 and do the calculations from step 3 to step 7. Check whether it has reached at the end of cycle, if so determine the root mean square error value, mean percentage of error and worst percentage of error over the complete patterns. Check whether the error is reasonable, if yes go to step 9 otherwise go to step3 and repeat the cycle from step 3 to step7.

Step 9: Stop the iteration and note the final weight values attached to the hidden and output layer neurons.

Step 10: Test the neural network model with the trained weight values. Determine the output values for the testing pattern and check whether the deviation from desired value is reasonably less. If no, try the back-propagation with the revised network by changing the number of neurons, altering learning rate parameter, altering momentum value and altering temperature value.

NEURAL NETWORK MODEL

An FEA approach has been used to evaluate the cutting forces and feed forces developed during the orthogonal cutting of Glass/Epoxy composite material for varied fibre percentage, fibre orientation and depth of cut (doc). These forces were used to train the network. In the network construction the number of hidden neurons, learning rate parameter, temperature are unknown values and these values are determined by trial and error. The network configuration of MLP usually consists of an input layer, one or more hidden layers and an output layer. It has been proved that one hidden layer is enough to approximate any continuous function [3]. More number of hidden layers proved to be counterproductive, it may cause slower convergence in the back propagation algorithm. Initially the number of hidden neurons increased from 2 to 10 and the network is trained for hidden neurons 2,4,6 and10 with the combination of learning rate parameter (η=1.2) and momentum factor (α=0.9). The convergence is checked for each hidden neuron, the minimum error occurred for the network constructed with 6 number of hidden neurons. This variation of error with number of neurons is shown in table.1. Thus the network configuration of 3-6-2 is used.

Table 1: Variation of error with neurons for glass/epoxy

No of neu-rons	Mean error	RMS error	Mean % of error	Worst % of error
4	0.001561	0.000006	3.41	22.06
6	**0.001529**	**0.000003**	**2.26**	**7.01**
8	0.002064	0.000007	3.26	18.77
10	0.001673	0.000004	2.91	15.31

When the steepest decent method is used, the amount of weight change along the error surface depends on η. Usually, with low order magnitude of η, there are only smaller changes to the synaptic weights in the network after each cycle. This results in smoother trajectory in the error-weight space. Selection of higher order η may results in faster convergence through larger changes in the synaptic weights, however the learning rate characteristics may not be a smooth trajectory and such values of η may lead to oscillation in the network performance. One attempt of increasing the speed of convergence, while minimizing the possibility of oscillation involves adding a term called momentum factor to the basic gradient decent formulation. In other words momentum is to magnify the learning rate for the flat regions of weight space where gradients are more or less constant, and to prevent oscillation.

$$\Delta w_{ji}(n+1) = \eta\,(\delta_{pj}\,O_{pi}) + \alpha\Delta w_{ji}\,(n).$$

η- Learning rate parameter.
α- Momentum.

Referring to the above equations, it can be seen that weight values at any given instant of cycle depends on its previous value and a factor influenced by η and α. In this network model α is maintained constant as 0.9.The learning rate parameter is varied from 0.2 to 1.3, the minimum error occurred for the learning rate parameter as 1.2. This effect of learning rate parameter over the error value is shown in table.2.

Table 2: Variation of error with learning rate parameter for glass/epoxy

Learning rate paramete r	Mean error	RMS error	Mean % of error	Worst % of error
0.2	0.00200	0.000008	3.67	17.02
0.4	0.001925	0.000007	3.47	16.53
0.6	0.001853	0.000006	3.46	15.99
0.8	0.001946	0.000006	3.64	15.61
1.0	0.001838	0.000006	2.45	12.69
1.1	0.001647	0.000004	3.10	16.08
1.19	0.001651	0.000003	2.78	10.01
1.2	**0.001529**	**0.000003**	**2.26**	**7.01**
1.21	0.001625	0.000004	2.78	11.21
1.3	0.001822	0.000005	2.99	15.85

Finally the effect of temperature value is checked, the change in temperature changes the sigmoid function. Temperature value of 1.0 gives minimum error value. This variation of error with temperature is depicted in table 3. Based on the above observations, following optimum parameters are selected.

Hidden nodes: **6**,
Learning rate parameter **1.2**,
Momentum factor: **0.9**.
Temperature: **1**

With the above parameters the network was trained for number of iterations. Initially high fluctuations were observed, after few cycles there is no much change in the error value. The network is stopped at that cycle, beyond which the network starts to over-learn and causes the error to increase once again. This effect of cycles over the error is illustrated in table.4. Optimum number of cycles for this network is observed to be 60000. Fig.2 illustrate the variation of mean percentage of error with the number of iterations.

Table 3: Variation of error with temperature for glass/epoxy

Temp	Mean error	RMS error	Mean % of error	Worst % of error
0.8	0.001375	0.000003	2.42	16.22
0.9	0.001651	0.000004	2.88	16.50
1.0	**0.001529**	**0.000003**	**2.26**	**7.01**
1.1	0.001500	0.000004	2.24	8.98

The trained weight values are used to test the model with the available desired values. Thus the network and FEA results are compared and it is observed that the maximum percentage of error occurred is in the order of 6 while the minimum percentage of error occurred is in the order of 0.93. This comparison is shown in table 5.

Table 4: variation of error with number of cycles for glass/epoxy

Cycles	Mean error	RMS error	Mean % of error	Worst % of error
1	0.20566	0.073409	328.8	3144.
100	0.07653	0.010809	95.90	326.5
1000	0.01191	0.000242	13.45	47.63
2000	0.003244	0.000015	5.11	15.59
3000	0.002465	0.000011	4.23	15.07
4000	0.002297	0.000010	3.98	14.39
5000	0.002200	0.000009	3.79	13.73
10000	0.002038	0.000007	3.25	13.36
20000	0.002196	0.000006	3.12	13.39
30000	0.002052	0.000006	2.89	11.72
40000	0.001941	0.000006	2.65	8.56
50000	0.001332	0.000003	2.29	7.73
60000	**0.001529**	**0.000003**	**2.26**	**7.01**
70000	0.002122	0.000008	3.89	16.02

Table 5: Comparison of network values and desired values

Fibre %	Fibre angle	DOC mm	Fc N/mm	Err %	Ft N/mm	Err %
30%	30^0	0.25	972.4	0.9	159.6	0.52
30%	45^0	0.25	119.4	6.3	16.4	6.4
30%	60^0	0.25	191.7	3.3	20.7	2.9
30%	75^0	0.25	620.8	1.4	51.5	1.53
40%	45^0	0.25	164.4	2.3	21.5	2.7
50%	45^0	0.25	202.1	3.1	27.9	3.4
60%	45^0	0.25	250.2	3.1	37.2	2.1
70%	45^0	0.25	301.4	2.3	47.2	3.1
30%	45^0	0.12	53.1	3.4	6.95	2.8
30%	45^0	0.50	226.14	2.6	36.2	2.7

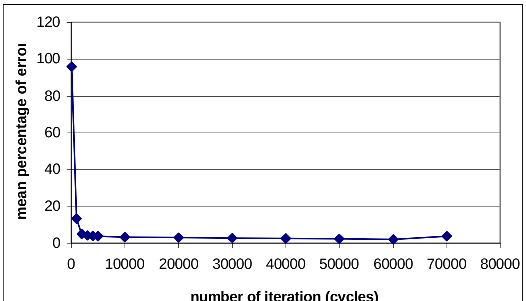


Fig.2Variation of mean error with the number of cycles.

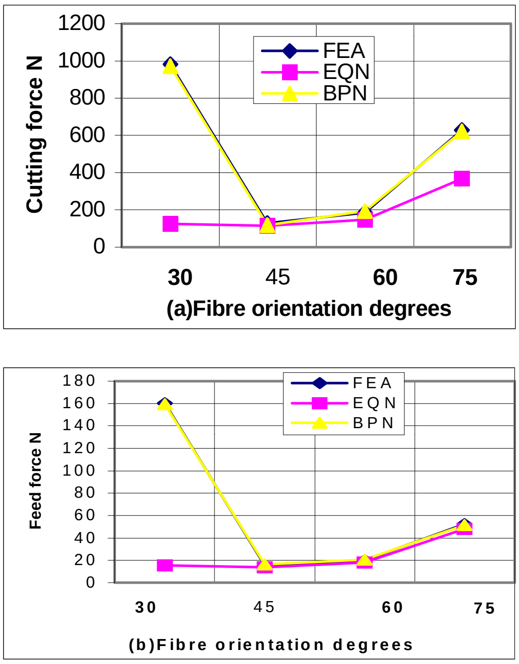


Fig.3.(a) & (b) Comparison of tool forces with fibre orientation

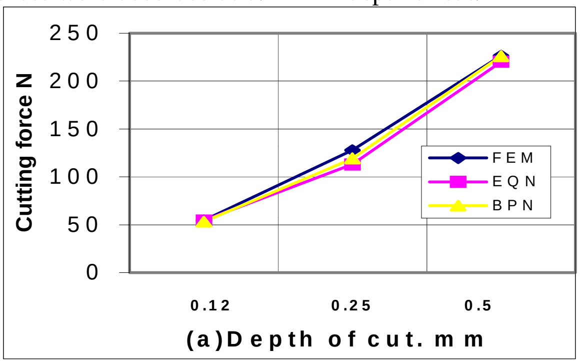
RESULTS AND DISCUSSION

Effect of fibre orientation

Fig.3 (a) and (b) illustrates the comparison of the cutting force (Fc) and feed force (Ft) values predicted by Merchant's equation, FEA model and neural network for various fibre orientation. Except at 30^0 fibre orientation the results of Merchants equation are well agreeing with other fibre orientations. The cutting forces are observed to be decreasing with the increase of fibre orientation up to 60^0 and then gradually increase with the fibre orientation. Maximum and minimum cutting forces were observed at 30^0 and 45^0 fibre orientation respectively.

Effect of depth of cut

Fig.4 (a) and (b) illustrates the comparison of cutting force (Fc) and feed force (Ft) values evaluated by Merchant's equation, FEA model and neural network for various depth of cuts. The results of Merchant's model are very well matching with the FEA and Neural results for 0.12 and 0.25mm depth of cut, however a slight variation in feed force was observed at 0.5mm depth of cut. A linear increase in cutting forces with respect to depth of cut was observed. Minimum cutting forces were observed at 0.12-mm depth of cut.



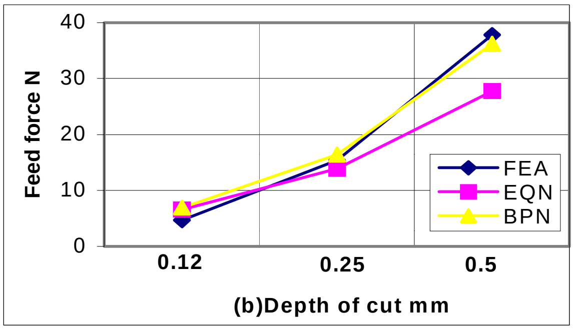
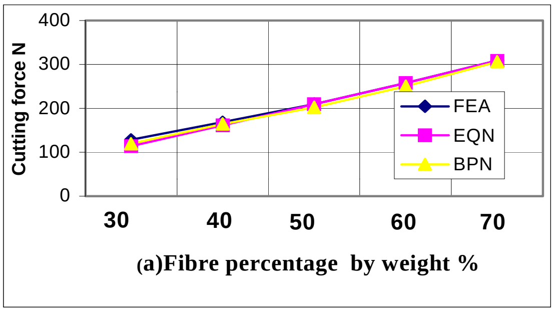


Fig. 4.(a) & (b) Comparison of Tool force with depth of cuts

Effect of fibre percentage.

Fig.5(a) and (b) illustrate the cutting force values predicted by Merchant's model, FEA and Neural Network analysis with respect to fibre percentage. The feed force values evaluated by Merchant's equation are exactly matching with the FEA and network results Though feed force values are showing good agreement with the FEA and Network results up to 50% fibre percentage, a slight variation was observed at 60 and 70% fibre content. In both the cases the trend was observed to be of increasing order, ie with the increase of fibre content the cutting forces were increased.



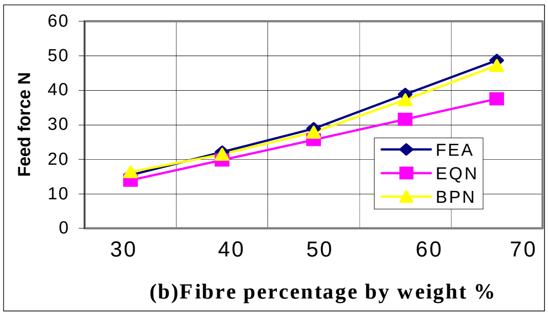


Fig.5.(a) & (b) Comparison of Tool forces on various fibre percentages

CONCLUSIONS

Though the neural network model was trained well and has given mean percentage of error in the order of 3, the network cannot determine cutting forces with negligible error or zero error. However the work carried out is a pioneer to update the model to make it as an error free model. For further improvement of the model the following suggestions are require to be implemented.

- To evaluate the cutting forces only three variables viz fibre percentage, fibre orientation and depth of cut were considered for the construction of neural network. As the model lacks generality, to improve the network the number of input neurons should be increased by considering other variables like tool angles, tool wear and surface roughness.

- Learning rate parameter is maintained constant throughout the iterations. At a given point on the error surface one weight may be dropping sharply while the other remains virtually constant..

- Another technique that has been suggested for the avoidance of local minima involves; injecting noise produced by a random generator may be quite large. As training proceeds the noise is gradually reduced and ultimately eliminated to achieve convergence.

- If the above suggestions have been considered for constructing a neural network, the model will act as a powerful tool for predicting the cutting forces with negligible error.

- The Neural network results have shown good agreement with the FEA results.

REFERENCES

Warren .Liao.T and Chen. L. J."Manufacturing Process Modeling and optimization based on Multi-Layer perceptron Network." Journal of manufacturing science and Engineering,.vol.120.pp.109-119.(Feb.1998)

Dornfeld. "A Neural Network sensor fusion for tool condition monitoring", Annals of the CIRP, Vol.39,. pp.101-105. (1990)

Dimal D. E.,.Lister P.M and Leighton. N .J. "Neural network solutions to the tool condition monitoring problem in metal cutting- A critical review methods". International Journal of Machine Tools Manufacturing, 37, , pp.1219-1241.(1997)

Yao, Y. L and .Fang, X "Assessment of chip forming patterns with tool wear progression in machining via neural networks", International Journal of Machine Tool Manufactures, 33, , pp.89-102. (1993)

Abhijit S. Pandya and Roberts B. Mag. "Pattern Recognition with Neural Networks in C++".

Valluru Rao a nd Haya griva Rao. "C++ Neural Networks and Fuzzy Logic".