ISSN: 0975-766X
CODEN: IJPTFI
Available Online through
www.ijptonline.com
Research Article

USING OF THREE TRAINING ALGORITHMS FOR MATERIAL REMOVE RATE (MRR) IN ELECTRICAL DISCHARGE MACHINING (EDM) FOR ANSI D2 STEEL BY NEURAL NETWORK: A COMPARATIVE STUDY

Amrut Ranjan Jena*[1], Raja Das[2], D.P.Acharjya[1]
School of Computer Sc. and Engineering, VIT University, Vellore, Tamil Nadu-632014.
School of Advanced Science, VIT University, Vellore, Tamil Nadu-632014.
Email:amrut.ranjan7@gmail.com

Received on 10-05-2016 *Accepted on 09-06-2016*

Abstract

In this work the comparison of the back propagations training algorithms of neural networks, mainly, Resilient back propagation, Conjugate gradient and Levenberg Marquardt methods are narrated. In this paper, material removal rates are predicted by comparing the effectiveness and efficiency of three training algorithms on the networks. Electrical Discharge Machining (EDM), the most non-traditional manufacturing procedures, is popular, due to its not requiring cutting tools and permits machining of hard, brittle, thin and complex geometry. Thus it is very popular in the field of modern manufacturing industries such as surgical components, nuclear industries aerospace. Based on the study and test results, although the Levenberg Marquardt has been found to be faster and having better performance than other algorithms in training, the Resilient back propagation algorithm has the best accuracy in testing period.

Keywords: Electro Discharge Machining, Material Remove Rate (MRR), Conjugate Gradient and Resilient Bck Propagation and Levenberg Marquardt.

1. Introduction:

Electrical Discharge Machining (EDM) is one of the significant process is often referred as the most non-conventional manufacturing process via electro thermal effect regardless of the work piece material [Pradhan2010]. EDM is a valuable and viable process in aerospace, automobile, mould making and surgical industries. The work piece and electrode are submerged inside the dielectric separated by a small gap; a spark is generated by the application of electrical energy. Due to the spark there is a localized temperature rise beyond 10000 degree centigrade which causes melting and even evaporation at the point of spark. The evaporated and molten material then flushed away the dielectric. Therefore, it is capable of manufacturing the material regardless of its hardness, strength and other capabilities, as long as the material is conductive. It is a reliable, affordable, and acceptable method for intricate and complex shapes, but also very complex and expensive too, that involves a combination of many fields of engineering. Hence, to understand the process modelling is required.

2. Literature Review

Since 1971several modelling attempts have been made to illustrate the EDM process based on electro thermal theory. But for predicting the behaviour of the EDM process, the conventional modelling techniques are found to be insufficient. The ANN can be used as a modelling tool, because the ANN can deliver justification to a large variety of problems with complex and uncertain data similar to EDM. Therefore, the use of ANN in modelling using responses obtained from experiments connecting different machining and materials conditions is gaining popularity. For predicting the behaviour of the EDM process, the conventional modelling techniques are found to be inadequate and the artificial neural network (ANN) can be used as a modelling tool. There are various ANN applications in EDM Mandal et al. (2007), Panda and Bhoi (2005), Wang et al. (2003), Gao et al. (2008) , Pradhan (2009) offered an response surface methodology and ANN predictive modelling using /P, *Ton*, and dielectric flushing pressure to predict overcut, MRR and tool wear rate . A close agreement was observed among the actual experiment, response surface methodology and ANN predictive results. Tsai and Wang (2001), Tsai and Wang (2001) have applied the radial basis function network **(**RBFN) and adaptive neural fuzzy inference systems (ANFIS) models have shown consistent results. Pradhan and Biswas (2010) in their study used two neuro-fuzzy models and a neural network model for predictions of MRR, tool wear rate, and radial overcut in die sinking EDM. The comparison results tell that the artificial neural network and the Neuro-fuzzy models are comparable in terms of speed and accuracy. Then a Neuro-fuzzy model and a regression model was developed to predict MRR, experiments were conducted with various levels of Ip, Ton and duty fraction. The model predictions were compared and found that the Neuro-fuzzy model has better predictive capability than the regression model Pradhan and Biswas (2009, 2008). These techniques have been used for modelling and optimization of various machining processes (Pradhan et al. (2010); Pradhan and Biswas(2011)).

3. Artificial Neural Network

An ANN is a biologically inspired computational model that processes information. ANN has been shown to be highly flexible modelling tool with capability of learning the mathematical mapping between input and output. ANN is formed from several layers of neurons. The input layer of neurons is connected to the output layers of neurons through one or more hidden layers of neurons. Initially ANN is trained and tested with experimental data to reach at an optimum topology and weights. A multilayer perception (MLP) is feed forward neural network with one or more hidden layers. During the training process ANN adjusts its weights to minimize the errors between the predicted result and actual output by using different back-propagation algorithms. The Back-Propagation Neural Network (BPNN) with n input nodes output nodes and a single hidden layer of m nodes are shown in Figure.1. Each interconnection between the nodes has a weight associated with it. The Transfer function of the hidden and output nodes are tan-sigmoid S (*) and linear respectively.

According to Fig. 1 the net input to the *j*th hidden neuron is given by

$$y_j(x) = \sum_{i=1}^{n} w_{1ji} + b_{1j}$$

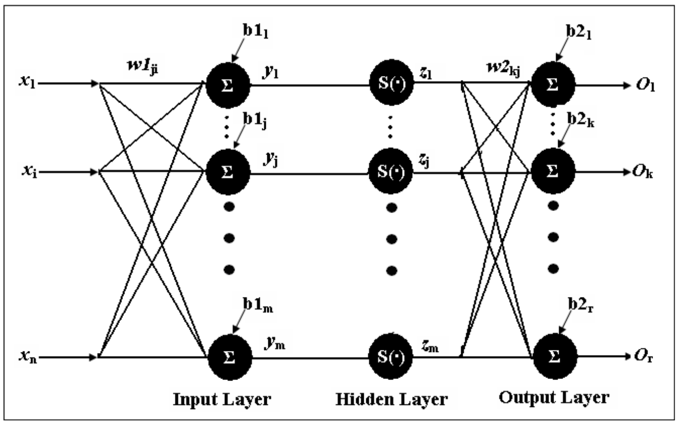

Figure-1: (Architecture of BPNN with single hidden layer).

Where w_{1ji} is the weight between the i*th* node of input layer and *j*th node of hidden layer and b_{1j} is the bias at *j*th node of hidden layer. The output of the *j*th hidden node is defined by

$$z_j(x) = \frac{2}{(1+\exp(-2x))-1}$$

Given an input vector x the output value $o_k(x)$ of the *k*th node of output layer is equal to the sum of the weighted outputs of the hidden nodes and the bias of the *k*th node output layer and is given by

$$o_k = \sum_{j=1}^{m} w_{2kj} z_j + b_{2k}$$

where w_{2kj} is the weight between the *j*th node of hidden layer and *k*th node of output layer b_{2k} is biasing term at the *k*th output node. The output of ANN is determined by giving the inputs and computing the output from various nodes activation and interconnection weights.

The output is compared to the experimental output and Mean Squared Error is calculated. The error value is then propagated backwards through the network and changes are made to the weights at each node in each layer by three different training algorithms.

3.1 ANN Training Algorithms:

The present work describes three different artificial neural network (ANN) training algorithms Levenberg-Marquardt conjugate gradient and resilient back propagation used in the study. This was done with a view to see which algorithm produces better results and has faster training for the application under consideration.

The objective of training is to reduce the global error E defined as

$$E = \frac{1}{P}\sum_{P=1}^{P} E_P$$

Where P is the total number of training dataset; and Ea is the error for Pth training data E_P is calculated by the following formula

$$E_\alpha = \frac{1}{2}\sum_{q=1}^{r}\left(O_q - t_q\right)^2$$

Where r is the total number of output nodes O_q is the network output at the qth output node and t_q is the target output at the qth output node.

In every training algorithm an attempt is made to reduce this global error by adjusting the weights and biases.

3.2 Levenberg- Marquardt (LM) Algorithms:

The approximated Hessian matrix $J^T J + \mu I$ is invertible; Liebenberg-Marquardt algorithm introduced another approximation to Hessian matrix stated as $H = J^T J + \mu I$, where J is the Jacobian matrix which contains first derivative of the network errors with respect to the weights and bias μ is always positive called combination coefficient and I is the identity matrix.

The Levenberg-Marquardt algorithm uses this approximation to the Hessian matrix in the following Newton-like update:

$$W_{K+1} = W_K - \left(J_K{}^T + \mu I\right)^{-1} J_K e_K$$

Where W is the weight vector and e is the error vector.

As the combination of the steepest descent algorithm and the Gauss-Newton algorithm the Levenberg-Marquardt algorithm switches between the two algorithms during the training process. When the combination coefficient μ is very small (nearly zero) Gauss-Newton algorithm is used. When combination coefficient μ is very large the steepest descent method is used. The Levenberg- Marquardt optimization technique is more powerful than the conventional gradient descent techniques.

4. Conjugate Gradient (CG) Algorithm:

The standard conjugate gradient method is to minimize the differentiable function E by generating a sequence of approximation W_{K+1} iteratively according to $W_{K+1} = W_K + \alpha_K d_K$

The scalar α_K is the step length known in neural network notation as learning rate. The step length α_K can be determined by line search techniques in the way that $E(W_K + \alpha_K d_K)$ is minimize along the direction α_K given W_K and d_K fixed. The standard conjugate gradient algorithm begins the minimization process with initial estimate W_0 and an initial search direction $d_0 = -\nabla E(W_0) = -g_0$. Each direction dk+1 is chosen to be linear combination of the steepest direction - g k+1 and the previous direction *dk*. We write $d_{K+1} = -g_{K+1} + \beta_K d_K$ where the scalar β_K is to be determined by the requirement that *dk* and d_{K+1} must fulfil the conjugate property. There are many formulae for the parameters *bk*. One of them is Fletcher-Reeves formula and is given by $\beta_K = \frac{g_K{}^T g_K}{g_{K-1}{}^T g_{K-1}}$. Conjugate gradient method has a second order convergence property without complex calculation of the Hessian matrix. A faster convergence established than lust order steepest descent approach.

4.1 Resi Bent Back- Propagation (RP) Algorithms:

The individual update value $\Delta ij(k)$ for each weight $w_{ij}(k)$ can be expressed according to the learning rule for each case based on the observed behaviour of the partial derivative during two successive weight-steps by the following formula:

$$\Delta_{ij} = \begin{cases} \eta^{+}\Delta_{ij}(k-1), if \frac{\partial E}{\partial w_{ij}}(k).\frac{\partial E}{\partial w_{ij}}(k-1) > 0 \\ \eta^{-}\Delta_{ij}(k-1), if \frac{\partial E}{\partial w_{ij}}(k).\frac{\partial E}{\partial w_{ij}}(k-1) < 0 \\ \Delta_{ij}(k-1) else \end{cases}$$

Where $0 < \eta^- < 1 < \eta^+$.

It is evident that whenever that partial derivative of the equivalent weight W_{ij} varies its sign which indicates that the last update was large in magnitude and the algorithm has skipped over local minima the update-value $\Delta_{ij}(K)$ is decreased by the factor η^-. If the derivative holds its sign the update-value will to some extent increase in order to speed up the convergence.

When the update-value for each weight is settled in the weight updates by a very simple rule.

$$W_{ij}(K+1) = W_{ij}(K) + \Delta W_{ij}(K)$$

Where

$$\Delta_{ij}(k) = \begin{cases} -\Delta_{ij}(k), if \dfrac{\partial E}{\partial w_{ij}}(k) > 0 \\ \Delta_{ij}(k), if \dfrac{\partial E}{\partial w_{ij}}(k) < 0 \\ 0, else \end{cases}$$

If the partial derivative changes sign that is the previous step was too large and the minimum was missed the previous weight-update is reverted:

$$\Delta W_{ij}(K) = -\Delta W_{ij}(K-1), if \frac{\partial E}{\partial W_{ij}}(K).\frac{\partial E}{\partial W_{ij}}(K-1) < 0$$

In order to avoid a double penalty of the update-value there should be no adaptation of the update value in the succeeding step. In practice this can be done by setting

$\dfrac{\partial E}{\partial W_{ij}}(K-1) = 0$ in the Δ_{ij} update rule above.

Hence the partial derivative of the errors must be accumulated for all training data. This indicates that the weights are updated only after the presentation of all of the training data.

It is noticed that resilient back-propagation is much faster than the standard steepest descent algorithm.

5. Result and Discussion

In the present BPNN model the input of the model are $T_{on}, T_{off}, I_p, \nu$ .The output is material removal rate (MRR).In this work a three layer BPNN is used having one intermediate hidden layer. Multilayer BPNN can have many hidden layers; however a single layer can also produce good result to approximate any complex non linear function.

Many experimental results confirmed that one hidden layer is enough for many forecasting problems. Here we used 810 neurons for hidden layer and 21neurons for LMPG and RP algorithms respectively for the given data set in table-1.

Exp. No.	T_{on}	T_{off}	I_p	ν
1	1	5	12	50
2	5	5	12	50
3	5	100	12	50
4	10	100	12	50
5	20	100	12	50
6	5	100	1	50
7	1	5	1	50
8	5	5	1	50
9	1	10	12	50
10	5	10	12	50
11	30	500	12	50
12	50	500	12	50
13	20	100	1	50
14	10	150	12	50
15	20	150	12	50
16	20	500	1	50
17	1	10	1	50
18	20	200	12	50
19	30	200	12	50
20	5	10	1	50
21	1	20	12	50
22	20	300	12	50
23	50	200	12	50
24	10	200	1	50
25	5	20	12	50
26	1	20	1	50
27	5	20	1	50
28	30	750	12	50
29	1	30	12	50
30	1	50	12	50
31	5	50	12	50
32	10	50	12	50
33	1	50	1	50
34	5	50	1	50
35	10	50	1	50
36	10	75	12	50
37	10	75	1	50
38	10	100	1	50
39	30	150	12	50
40	10	150	1	50
41	20	150	1	50
42	10	200	12	50
43	1	30	1	50
44	20	500	12	50

(Table.1)

The ANN algorithms were compared according to mean squared errors (MSE) and mean percentile error (MPE) criteria. These criteria are defined as for each combination ANN was trained using three different algorithms that is LM, CG and RP. After training was over the weights were saved and used to test the each network performance on test data. The ANN results were transformed back to the original domain and MSE was computed. The iterations were stopped when the difference between two epochs was too small. The experimental results and predicted results of Ray the LM, CG and RP were plotted on the same scale as shown in the Figure 2.

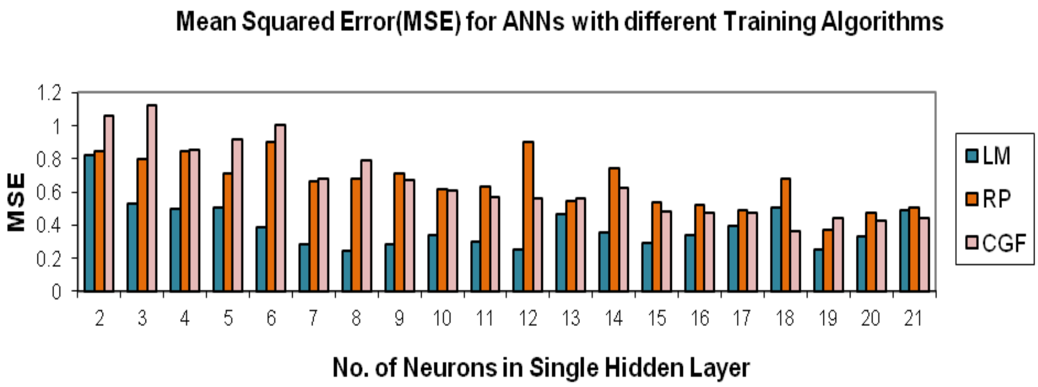

Figure-2

The regression analysis for ANN by different algorithms are given bellow.

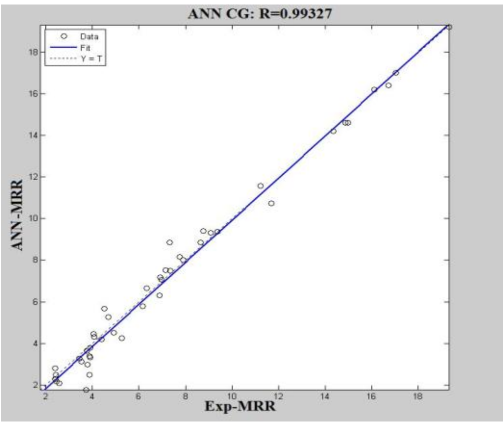

Figure-3: Regression analysis for ANN by CG.

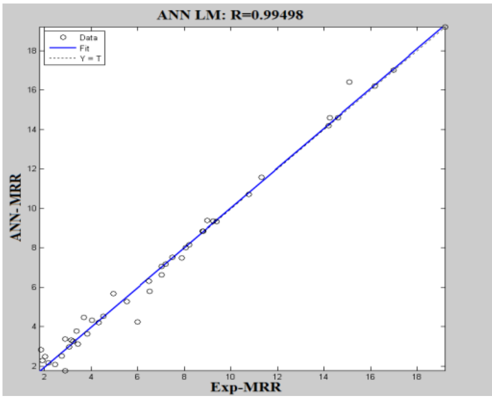

Figure-4Regression analysis for ANN by LM.

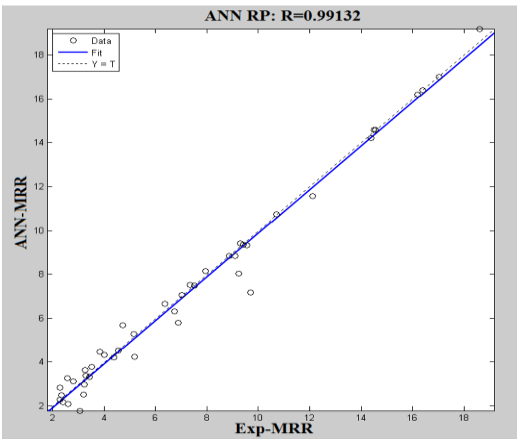

Figure-5: Regression analysis for ANN by RP.

The performance of the three ANN algorithms are calculated with a special reflection to their regression plot and given in the figure number 3, 4, and 5 respectively. The Better R value was given by LM algorithm and the figure 6 shows the comparison of ANN with experimental values.

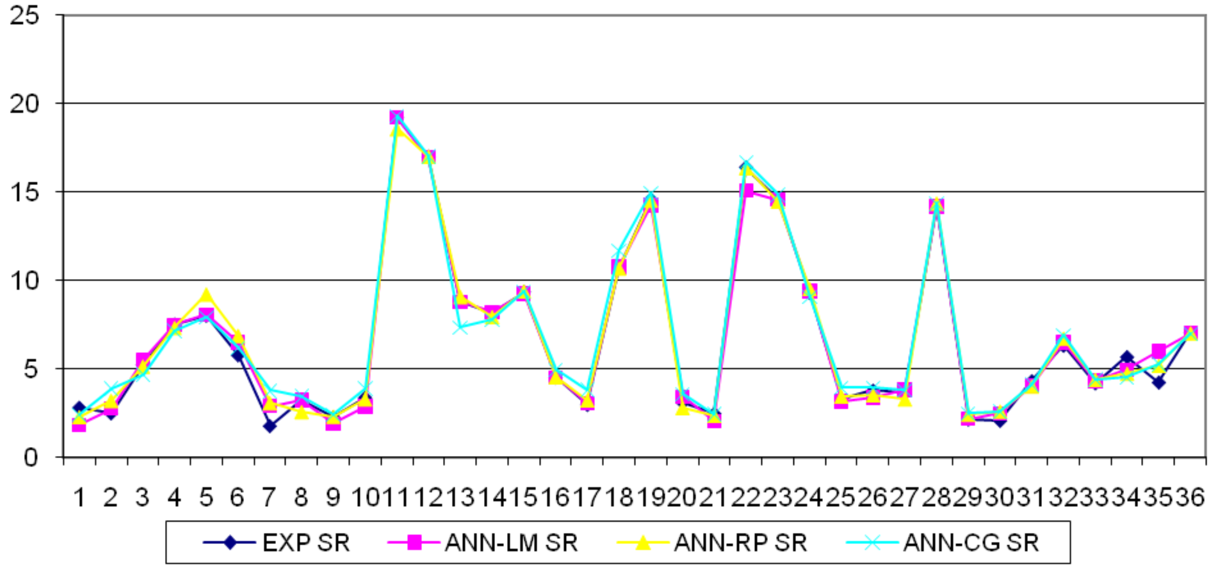

Figure-6: (comparison of ANN with experimental values).

ANNs are compared separately with results obtained by experiments and the average absolute error obtained for all the three networks. ANNs with CG and RP models are poorer in predicting Ra. The test result accuracy measured in terms of MAE and MPE for nine test data are given in table.2.

Table.2 (MAE and MPE of testing data for training algorithms LM, CG and RP).

	Number of nodes in hidden single layer	**Mean Absolute Error(MAE)**	**Mean Percentile Error(MPE)**
LM	11	0.471427066	9.67175711827
CG	19	0.868739301	26.0511400982
RP	21	0.7403860527	10.6566390978

Conclusion

In this work three artificial training algorithms LM, CG and RP were applied for the prediction of material removal rate of the Wired Electrical discharge machined surface. This work explains that the prediction of material removal rate is possible through the use of LM, CG and RP based neural network algorithms. The results obtained from widespread experiments conducted on ANSI D2 steel work piece materials with diverse machining parameters using copper electrode are compared and validated with the prediction. It was found to be close correlation with the experimental results. It was also noticed that the LM model is quite analogous with CG and RP for material removal rate. The LM network demonstrated a slightly better performance compared to other models. And also LM model predicted quite faster than the error goal reached in only 11 epochs while CG required 19 epochs and RP required 21 epochs. At last but not the least the material removal rate of EDMed surface can be predicted by the above models with reasonable accuracy.

Acknowledgments

The authors wish to acknowledge Dr. Mohan Kumar Pradhan of Department of Mechanical Engineering, Maulana Azad National Institute of Technology, Bhopal, India, for support towards the execution of this study.

References

1. Gao, Q., Zhang, Q., Su, S., Zhang, J., and Ge, R , Prediction models and generalization performance study in electrical discharge machinings. *Applied Mechanics and Materials,* 2008, 1012:677-681.
2. Anitha J, Das R, Pradhan M.K, Performance comparison of neural network training algorithm in modelling of surface roughness in EDM,*proc. of ICME*, 2014 ,pages 303-311,MANIT, Bhopal.
3. Mandal, D, Pal, S. K., and Saba.P., Modeling of electrical discharge machinings process using back propagation neural network and multi-objective optimization using non-dominating sorting genetic algorithm-II. *Journal of Materials Processing Technology,* 2007, 186:154-162.
4. Panda, D. K and Bhoi, R. K., Artificial neural network prediction of material removal rate in electro-discharge machining. *Materials and Manufacturing Processes,* 2005, 20:645-672.
5. Pradhan, B.B.and Bhattacharyya, B., Modelling of micro-electrodischarge machining during machining of titanium alloy Ti-6Al.-4V using response surface methodology and artificial neural network algorithm. *Proc. of the Institution of Mechanical Engineers, Part B.' Journal of Engineering Manuf.,* 2009, 223(6):683-693.
6. Pradhan, M. and Biswas, C., Neuro-fuzzy and neural network-based prediction of various responses in Electrical Discharge Machiningof AISI D2 steel - NF and NN based prediction of responses in EDM of D2 steel. */nternational Journal of Advanced Manufacturing Technology,* 2010, 50:591-610.
7. Pradhan, M. K. and Biswas, C. K., Modeling of machining parameters for MRR in EDM using response surface methodology. *Proc. of the National Conference on Mechanism Sciencand Technology.' From Theory to Application*, 2008, pages 535-542, NIT, Hamirpur, India.
8. Pradhan, M. K. and Biswas, C. K., Neuro-fuzzy model on material removal rate in Electrical Discharge Machiningin AISI D2 steel. In *Proceedings of the 2nd International & 23rd All IndiaManufacturing*

Technology, Design and Research Conference (AIMTDR) Conference, 2008,volume 1, pages 469-474,, IIT Madras, Chennai, India.

9. Pradhan, M K and Biswas, C. K., Neuro-fuzzy model and regression model a comparison study of MRR in Electrical Discharge Machiningof D2 tool steel. */nternational Journal of Mathematical, Physical and Engineering Sciences, WASET*, 2009, 3:48-53.
10. Pradhan, M K and Biswas, C. K., Investigation in to the effect of process parameters on surface roughness in EDM of AISI D2 steel by response surface methodology. *International Journal of Precision Technology, Vol. 2, No. 1*, 2011,pp-*'64-80, 2011*, 2:64-80.
11. Pradhan, M K and Biswas, C. K., Multi-response optimisation of EDM AISI D2 tool steel using response surface methodology. *International Journal of Machining and Macht.nability of Materials (/JMMM)*, 2011, 9:66-85.

Corresponding Author:

Amrut Ranjan Jena*,

Email: *amrut.ranjan7@gmail.com*