Spark resistance curve fitting using non-linear regression and kerf loss, slicing rate(SR) & thermal damage prediction using artificial neural network in silicon WEDM.

Ravi Sharma

Electrical Engineering ,Indian Institute Of Technology Bhubaneswar, Jatani ,Odisha 752050.

S. V. Kulkarni

Professor, Department of Electrical Engineering, Indian Institute of Technology Bombay, Mumbai,

Maharashtra 400076

Abstract

Increasing demand of electricity has been driving the world towards non-conventional sources of energy since last few decades. Photovoltaic(PV) systems are used to convert solar energy into electricity. But the use of the solar power system is limited due to its high cost. Effective slicing of silicon in solar power systems can minimize the cost. The WEDM can play an important role in effective slicing of silicon in PV industries. However, the WEDM is developed mainly for metals. To adapt it for cutting semiconductors like silicon, the nature of spark between silicon and steel needs to be understood. The spark resistance for cutting silicon and steel is modelled and compared in this study. To find the dependency of spark resistance on current and time, two non-linear equations are chosen to model spark resistance. Spark resistance dependency on time and current is compared in case of silicon and steel. Non-linear regression is used to model the spark resistance from available experimental data. Two models are compared based on the adjusted R-squared values. To get optimum value of input parameters to achieve minimum kerf loss, a neural network modelled is designed. Backpropagation algorithm is used to design the neural network model.

keywords: Normal equation, Non-Linear Regression, feature scaling, backpropagation

Abbreviations

Abbreviations

WEDM	Wire Electrical Discharge Machining
ANN	Artificial Neural Network
SE	Square Error
PV	Photovoltaic
SR	Slicing Rate

1 Introduction

1.1 Background

Wire electrical discharge machining is the most widely-used non-traditional machining process. It is used to get the desired shape of a metallic materials. Advantages of WEDM over other methods like miling are many. Any material that can conduct electricity can be processed in this machining technique irrespective of its hardness. The principle of WEDM is to use the eroding effect of controlled electric spark discharges on the electrodes. Spark is created between wire and workpiece, where wire is considered as cutting tool which continuously circulates on a programmed path. Problem of erosion of tool does not occur in WEDM because of continuous circulation of wire unlike other electric discharge machining processes. There is no physical contact between wire and workpiece during complete cutting process. Sparks are created in dielectic. Voltage between the wire and workpiece is kept constant, wire is slowly moved towards workpiece. As a result, electric field in the dielectric increases until it reaches the required value of breakdown. Breakdown occurs at the point where distance between wire and workpiece is minimum. When breakdown occurs, some region of dielectric is ionised which is called as plasma channel. During formation of plasma channel, current increases and voltage falls rapidly. Ions in the plasma channel strikes with the workpiece surface which causes strong heating of workpiece material. Small molten metal pool is formed on workpiece surface. When discharge is over, current and voltage are shut down. After discharge, dielectric plays an important role, it cools down electrodes and solidifies molten metal and flushes it away as shown in Fig 1. This is a critical phase of WEDM in which gap between wire and workpiece is cleaned otherwise small particles will stay in the gap and conductivity of dielectic will increase which will causes bad or poor control of machining process. The quality of machining is directly dependent on voltage, current, discharge time. Sparks with more current removes more material, but it increases surface roughness. The process of WEDM is shown in Fig.1

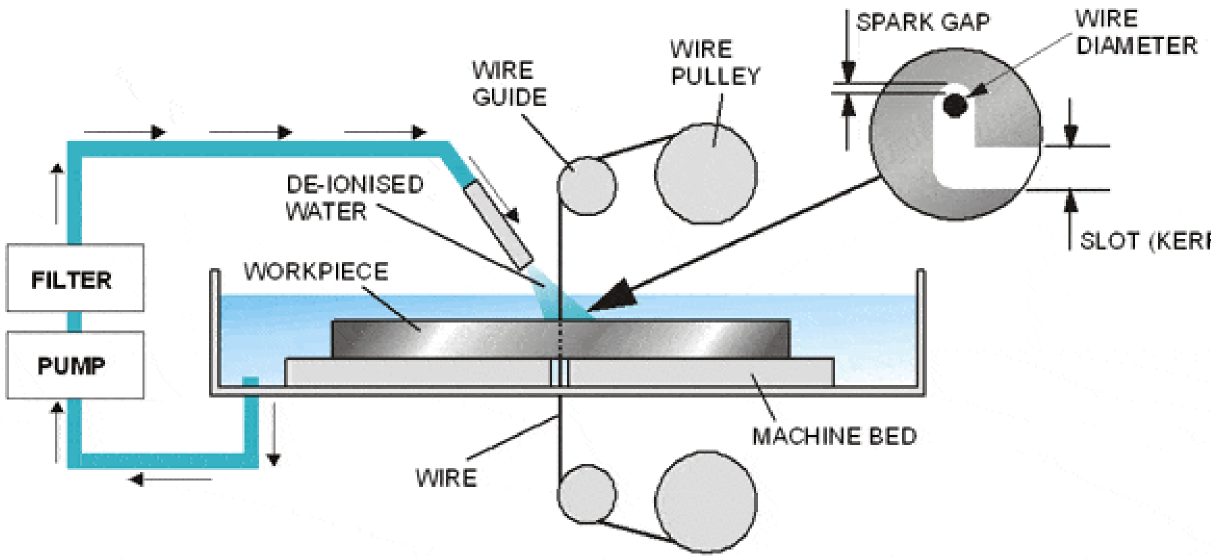


Fig 1: WEDM [6]

1.2 **Objective of the research**

This research work will create background for further research in silicon WEDM. Method of predicting spark resistance is discussed in this research for steel and silicon. Accuracy of method is verified using R-squared. Slicing rate, kerf loss, thermal damage is also predicted using a model designed using artificial neural network.

1.3 **Statement of the Problems**

Main goal of manufacturers is to get optimal productvity in WEDM process. Effective silicing of silicon will lower the cost of manufacturing of photovoltaic cells. But due to its stochastic nature, even highly-skilled operator is unable to get high productivity. So, study of its stochastic nature is required. In this research, stochastic nature of spark resistance of steel and silicon is examined. Kerf loss, Slicing rate, thermal damage are also predicted using ANN model.

2 **Methodology**

2.1 **Available data**

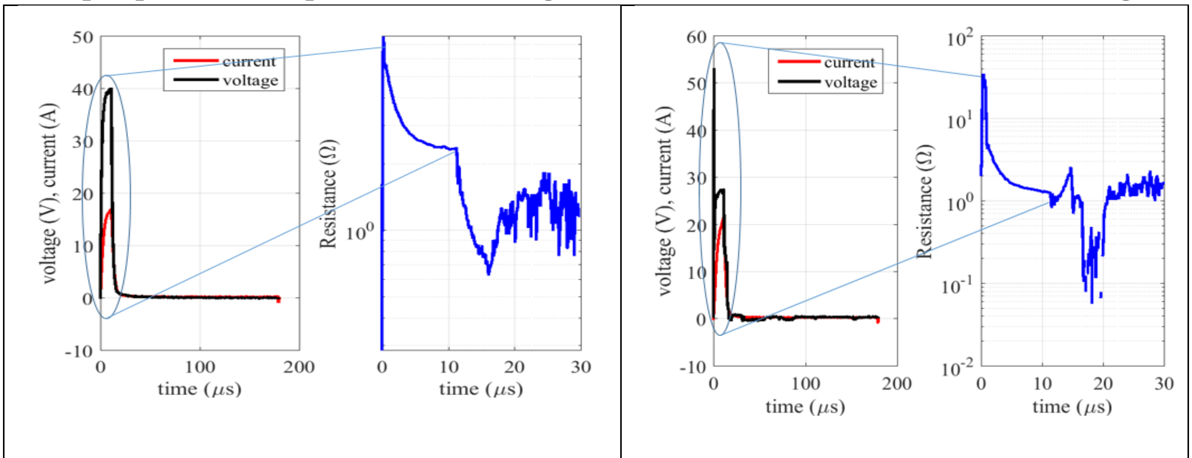
Experiments are done in IIT BOMBAY by varying five input parameters in WEDM machine. These parameters are Voltage, Current, T_{ON}, T_{OFF} and Wire-speed. These orthogonal arrays are shown in Table 1. For each of these 18 experiments several hundreds of voltage and current pulses are available in .CSV format.

Table 1: L_{18} orthogonal arrays

Voltage(V)	Current(A)	Ton	Toff	Wirespeed(rpm)
75	2	12	16	0
75	2	14	18	1
75	2	18	20	3
75	4	12	16	1
75	4	14	18	3
75	4	18	20	0

75	6	12	18	0
75	6	14	20	1
75	6	18	16	3
110	2	12	20	3
110	2	14	16	0
110	2	18	18	1
110	4	12	18	3
110	4	14	20	0
10	4	18	16	1
110	6	12	20	1
110	6	14	16	3
110	6	18	18	0

- Sample pulse for experimental voltage, current and resistance is shown in Fig.2



a) Silicon

b) Steel

Fig 2: Voltage, current and resistance for 1 pulse

- The non-linear nature of spark resistance can be seen in Fig 2.

2.2 Concepts used for spark resistance calculation

2.2.1 Proposed model

Several models [1] [3] are proposed to describe the behavior of spark resistance. The equation proposed for spark resistance ($\boldsymbol{R}$) is a non-linear function as shown in equation (1).

$$\boldsymbol{R} = \boldsymbol{k}\boldsymbol{i}^{\boldsymbol{m}} \tag{1}$$

Here, it is assumed that $\boldsymbol{k}$ incorporates all the parameters that are considered constant during the experiment. [1] [3]

Variable $\boldsymbol{i}$ is the current in sparking zone.

2.2.2 Linearization of proposed equation

Equation (1) is non-linear in nature. To apply linear regression, it is first converted into a linear equation by taking natural log on the both sides.

$$\mathbf{ln}(\boldsymbol{R}) = \mathbf{ln}(\boldsymbol{k}) + \boldsymbol{m} \times \mathbf{ln}(\boldsymbol{i}) \tag{2a}$$

$$\boldsymbol{y} = \boldsymbol{a_0} + \boldsymbol{a_1}\boldsymbol{x_1} \tag{2b}$$

Now, after taking natural log, equation (2a) is similar to linear equation as shown in equation (2b).

2.2.3 Optimization of coefficients $\boldsymbol{a_0}$ and $\boldsymbol{a_1}$

Normal Equation [2]: It is an analytical solution to the linear regression problem with a least-squares cost function.

Solution for coefficients is determined by equation (3) and equation (4) [2]:

$$\boldsymbol{\theta} = (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-\mathbf{1}}\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{y} \tag{3}$$

Consider the resistance data has `$\boldsymbol{m}$' training examples.

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{1} & \boldsymbol{x}_{\mathbf{1}}^{(\mathbf{1})} \\ \mathbf{1} & \boldsymbol{x}_{\mathbf{1}}^{(\mathbf{2})} \\ . & . \\ . & . \\ \mathbf{1} & \boldsymbol{x}_{\mathbf{1}}^{(\boldsymbol{m})} \end{bmatrix} \quad ; \quad \boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}^{(\mathbf{1})} \\ \boldsymbol{y}^{(\mathbf{2})} \\ . \\ . \\ \boldsymbol{y}^{(\boldsymbol{m})} \end{bmatrix} \tag{4}$$

where $\boldsymbol{x}_j^{(i)}$ is the jth feature value for ith training example and $\boldsymbol{y}^{(i)}$ is corresponding value of output for ith training example.

2.2.4 **Extracting coefficients**

Theta vector contains the optimized values of coefficients of equation (2a) as shown in equation (5a).

$$\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{ln}(\boldsymbol{k}) \\ \boldsymbol{m} \end{bmatrix} \tag{5a}$$

$$S0, \quad \boldsymbol{k} = \boldsymbol{e}^{\boldsymbol{\theta}(\mathbf{1})} \quad and \quad \boldsymbol{m} = \boldsymbol{\theta}(\mathbf{2}) \tag{5b}$$

2.2.5 **Model verification**

R-squared and Adjusted R-squared are calculated in equation 6, 7, 8 and 9 to check the performance of the model.

Correlation between predicted (*f*) and experimental (*y*) value of independent variable is determined by equation (6a) [2]

$$\boldsymbol{Cor}(\boldsymbol{y},\boldsymbol{f}) = \frac{\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{m}} (\boldsymbol{y}^{(\boldsymbol{i})} - \bar{\boldsymbol{y}})(\boldsymbol{f}^{(\boldsymbol{i})} - \bar{\boldsymbol{f}})}{\sqrt{\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{m}} (\boldsymbol{y}^{(\boldsymbol{i})} - \bar{\boldsymbol{y}})^{\mathbf{2}} \quad \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{m}} (\boldsymbol{f}^{(\boldsymbol{i})} - \bar{\boldsymbol{f}})^{\mathbf{2}}}} \tag{6a}$$

$$\boldsymbol{R}^{\mathbf{2}} = [\boldsymbol{Cor}(\boldsymbol{y},\boldsymbol{f})]^{\mathbf{2}} \tag{6b}$$

Coefficient of determination ($\boldsymbol{R}^2$) is the square of correlation coefficient [2]

$\boldsymbol{R}^2$ is also given by equation 7 and equation 8.

$$\boldsymbol{SS_{res}} = \sum_{i=1}^{m} (\boldsymbol{y}^{(i)} - \boldsymbol{f}^{(i)})^2 \tag{7a}$$

$$\boldsymbol{SS_{total}} = \sum_{i=1}^{m} (\boldsymbol{y}^{(i)} - \bar{\boldsymbol{y}})^2 \tag{7b}$$

$\bar{y}$ = mean value of y ; $f^{(i)}$ is the predicted value by model for i^{th} training example.

$$\boldsymbol{R^2 = 1 - SS_{res}/SS_{total}} \tag{8}$$

Here, if $\boldsymbol{SS_{res} = SS_{total}}$ then $\boldsymbol{R^2 = 0}$ which implies that there is no linear relation between $\boldsymbol{y}$ and parameters (independent variables). [2]

Linear dependency of $\boldsymbol{y}$ on parameters increases then $\boldsymbol{SS_{res}}$ approaches 0 and $\boldsymbol{R^2}$ approaches 1. [2]

$$\boldsymbol{AdjustedR^2 = 1 - (1 - R^2)\frac{n-1}{n-p-1}} \tag{9}$$

$\boldsymbol{n}$ = number of samples and $\boldsymbol{p}$ = number of parameters.

2.3 Concepts for neural network

2.3.1 Available data

Available data for slicing rate, kerf loss, thermal damage is shown in Table 2. This data is also obtained from experiments done in IIT Mumbai.

Table 2: Slicing rate, kerf loss and thermal damage

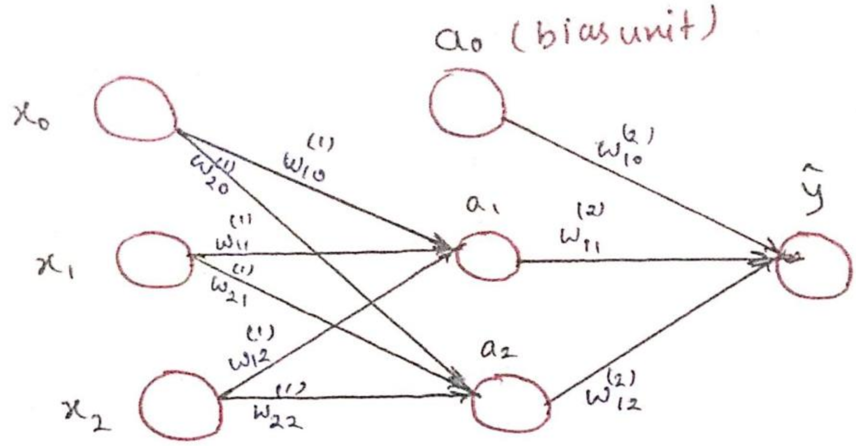
S.No	OV (volts)	SV (volts)	Ton (μs)	Toff (μs)	Slicing Rate (mm/min)	Kerf-loss (μm)	Thermal Damage (μm)
1	65	43	0.6	9	0.510309278	145.5179487	11.942575
2	68	41	0.5	7	0.68989547	143.5534188	10.9685
3	68	45	0.5	11	0.506393862	149.5491453	23.014875
4	62	41	0.5	11	0.440326168	145.892735	19.100375
5	71	43	0.4	9	0.719128329	143.5166667	11.63395
6	71	43	0.4	13	0.723507917	141.9692308	18.6725
7	71	43	0.6	9	0.719128329	150.2854701	13.063875

8	65	39	0.4	13	0.639397201	143.5431624	12.0793
9	65	39	0.6	13	0.639397201	136.4692308	11.492475
10	68	41	0.5	11	0.675767918	141.3867521	15.87175
11	68	37	0.5	11	0.887892377	131.6760684	12.0286
12	68	41	0.3	11	0.677309008	141.2029915	13.821925
13	65	43	0.4	9	0.453781513	146.0188034	23.0932
14	65	39	0.4	9	0.623294858	144.9213675	12.9174
15	71	43	0.6	13	0.68989547	147.2747863	23.34995
16	65	39	0.6	9	0.62394958	139.3102564	11.0496
17	71	39	0.4	9	0.914634146	133.4525641	11.098825
18	68	41	0.7	11	0.663716814	140.8970085	16.20405
19	71	39	0.6	9	0.630252101	135.1111111	21.829075
20	68	41	0.5	11	0.652883569	141.3867521	17.52175
21	65	43	0.4	13	0.442151805	146.2418803	29.496775
22	68	41	0.5	15	0.657894737	139.1645299	17.428475
23	68	41	0.5	11	0.658616905	141.3867521	16.88425
24	68	41	0.5	11	0.665158371	141.3867521	15.83425
25	65	43	0.6	13	0.435233161	144.2773504	23.4759
26	74	41	0.5	11	0.984771574	137.4209402	12.4176
27	71	39	0.6	13	0.905132193	130.4991453	17.13445
28	71	39	0.4	13	0.913657771	130.3038462	10.8142

2.3.2 **Model representation**

Artificial neural network models are used to predict the complex non-linear relationship between input parameters and output parameters. A ANN model [4] can be modelled as shown in Fig 3.

each node is representing a neuron. In th
figure shown there are 3 input neurons & 1 outp
neuron.



=) $w_{ij}^{(l)}$:- weight that is connecting i^{th} unit o
layer $(l+1)$ to j^{th} unit of layer (l

=) $a_1 = \phi(x)$
$= \phi(x_0 w_{10}^{(1)} + x_1 w_{11}^{(1)} + x_2 w_{21}^{(1)})$

=) $\phi(x)$ is called activation function.

Fig 3: Model representation

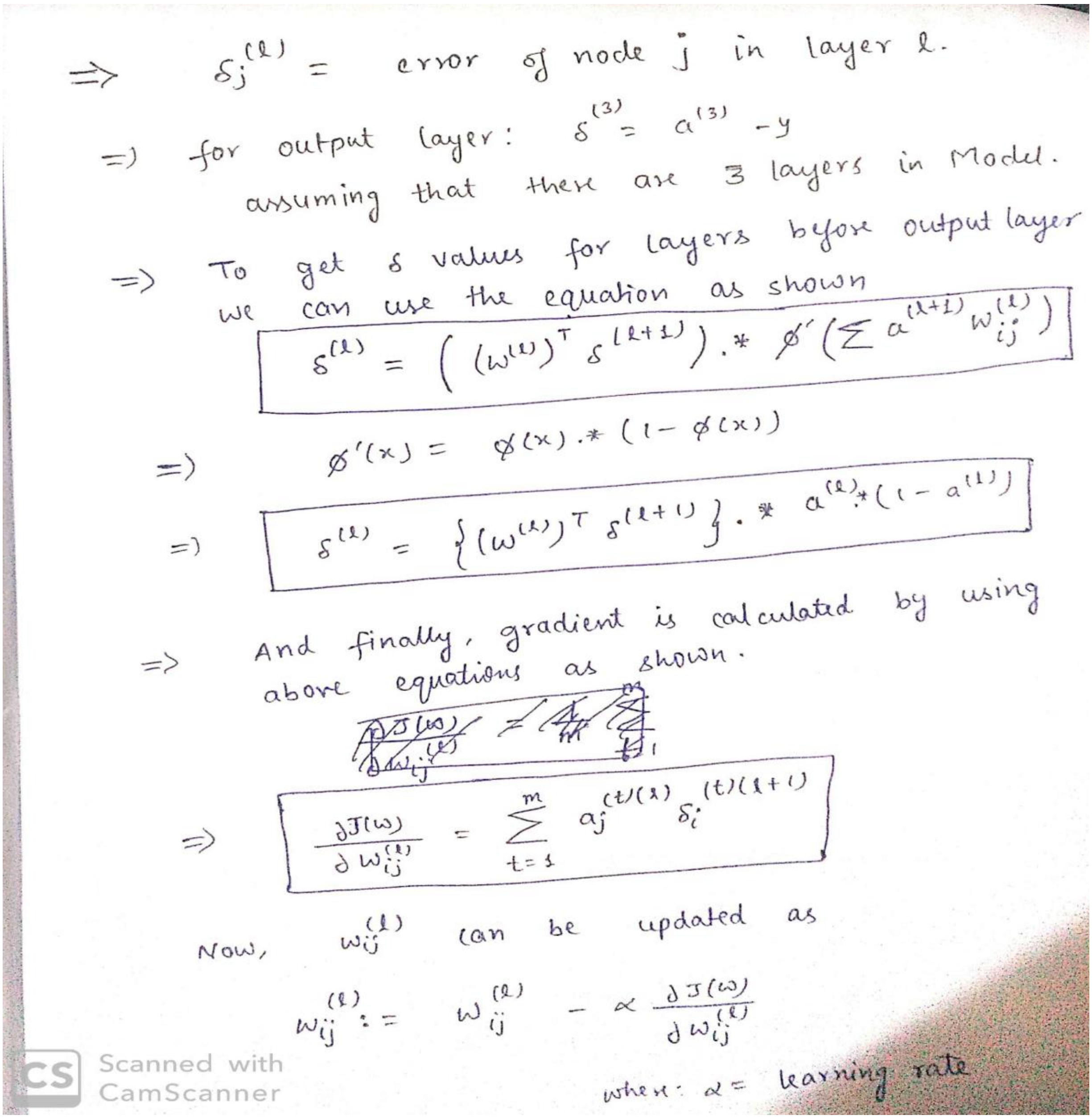
2.3.3 **Cost (Error) function**

Available data is of regression type. So the error function *(J(w))* used is 'square error'.

$$J(\omega) = (y_i - y_{pred_i})^2$$

2.3.4 **Backpropagation algorithm**

It is used to minimize cost function J*(w)* using an optimal set of weights. Backpropagation calculates the gradient term that is used to update weights(w_{ij}). To calculate the gradient term, error (δ) of every node in each layer is calculated [4]. Evaluation of backpropagation is shown in Fig 4.



⇒ $\delta_j^{(l)}$ = error of node j in layer l.

⇒ for output layer: $\delta^{(3)} = a^{(3)} - y$

assuming that there are 3 layers in Model.

⇒ To get δ values for layers before output layer we can use the equation as shown

$$\boxed{\delta^{(l)} = \left((w^{(l)})^T \delta^{(l+1)} \right) .* \phi'\left(\sum a^{(l+1)} w_{ij}^{(l)}\right)}$$

⇒ $\phi'(x) = \phi(x) .* (1 - \phi(x))$

⇒ $$\boxed{\delta^{(l)} = \{(w^{(l)})^T \delta^{(l+1)}\} .* a^{(l)} .* (1 - a^{(l)})}$$

⇒ And finally, gradient is calculated by using above equations as shown.

$$\cancel{\frac{\partial J(w)}{\partial w_{ij}^{(l)}} = \frac{1}{m} \sum_{t=1}^{m}}$$

⇒ $$\boxed{\frac{\partial J(w)}{\partial w_{ij}^{(l)}} = \sum_{t=1}^{m} a_j^{(t)(l)} \delta_i^{(t)(l+1)}}$$

Now, $w_{ij}^{(l)}$ can be updated as

$$w_{ij}^{(l)} := w_{ij}^{(l)} - \alpha \frac{\partial J(w)}{\partial w_{ij}^{(l)}}$$

where: α = learning rate

Fig 4: Backpropagation Algorithm

3 **Results and discussion**

3.1 **Spark resistance**

After fitting the equation for all 18 experiments as shown in Table (1). Different values of ***k***, ***m*** are obtained for each array as shown in Table 3.

Table 3: ***k*** and ***m*** values for silicon and steel

	k		***m***	
EXP NO.	SILICON	STEEL	SILICON	STEEL
1	10.50	73.26	-0.52	-1.36
2	6.95	54.81	-0.36	-1.32
3	7.14	65.91	-0.37	-1.37
4	33.53	77.14	-0.80	-1.29
5	22.83	79.05	-0.70	-1.33
6	9.22	56.68	-0.41	-1.21
7	10.57	57.05	-0.48	-1.15
8	8.78	67.17	-0.42	-1.24
9	8.37	75.31	-0.44	-1.26
10	35.16	117.03	-0.89	-1.45
11	16.70	52.89	-0.65	-1.25
12	21.36	58.90	-0.74	-1.28
13	33.79	86.39	-0.84	-1.26
14	15.27	56.98	-0.61	-1.18
15	17.13	52.84	-0.65	-1.15
16	18.48	78.05	-0.66	-1.20
17	10.02	84.32	-0.52	-1.24
18	9.05	61.84	-0.49	-1.15

- Some authors suggested that the radius of plasma channel depends on time [1] [3]. So, to take this into consideration, time is also included as a parameter in equation (1) and time dependency of spark resistance is observed. It is shown in equation (10) · [1] [3]

$$\boldsymbol{R = k i^{m} t^{n}} \tag{10}$$

Here, feature scaling is used to normalise the range of time vector as shown in equation (11).

$$\boldsymbol{t^{(i)}} = \frac{\boldsymbol{t^{(i)}}}{\boldsymbol{max(t) - min(t)}} \tag{11}$$

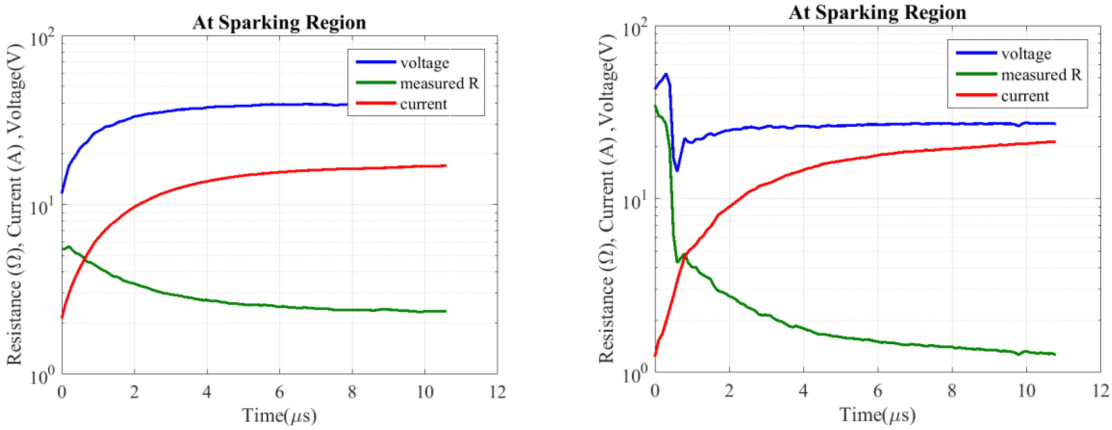
$\boldsymbol{t^{(i)}}$ is the time corresponding to ith training example

Using this model, different values of $\boldsymbol{k}$, $\boldsymbol{m}$ *and* $\boldsymbol{n}$ are obtained for each array of L_{18} as shown in Table 4.

Table 4: $\boldsymbol{k}$, $\boldsymbol{m}$ and $\boldsymbol{n}$ values for steel and silicon

	$\boldsymbol{k}$		$\boldsymbol{m}$		$\boldsymbol{n}$	
EXP NO.	SILICON	STEEL	SILICON	STEEL	SILICON	STEEL
1	5.89	297.66	-0.32	-1.80	-0.10	0.40
2	3.53	132.55	-0.11	-1.62	-0.14	0.28
3	2.67	186.24	-0.02	-1.72	-0.19	0.
4	36.06	242.24	-0.82	-1.61	0.01	0.35
5	88.09	273.58	-1.17	-1.70	0.28	0.42
6	13.59	225.31	-0.54	-1.61	0.07	0.40
7	7.13	207.49	-0.37	-1.50	-0.07	0.36
8	19.64	162.67	-0.67	-1.49	0.15	0.27
9	92.48	191.22	-1.16	-1.51	0.41	0.30
10	16.19	581.86	-0.64	-1.91	-0.14	0.46
11	32.56	202.33	-0.88	-1.67	0.14	0.38
12	25.55	359.86	-0.80	-1.84	0.04	0.50
13	28.91	254.24	-0.79	-1.55	-0.03	0.31
14	37.02	286.41	-0.88	-1.63	0.17	0.43
15	27.16	311.11	-0.79	-1.63	0.08	0.44
16	5.53	290.89	-0.33	-1.53	-0.22	0.36
17	7.72	274.02	-0.44	-1.56	-0.05	0.36
18	38.81	373.07	-0.89	-1.61	0.25	0.46

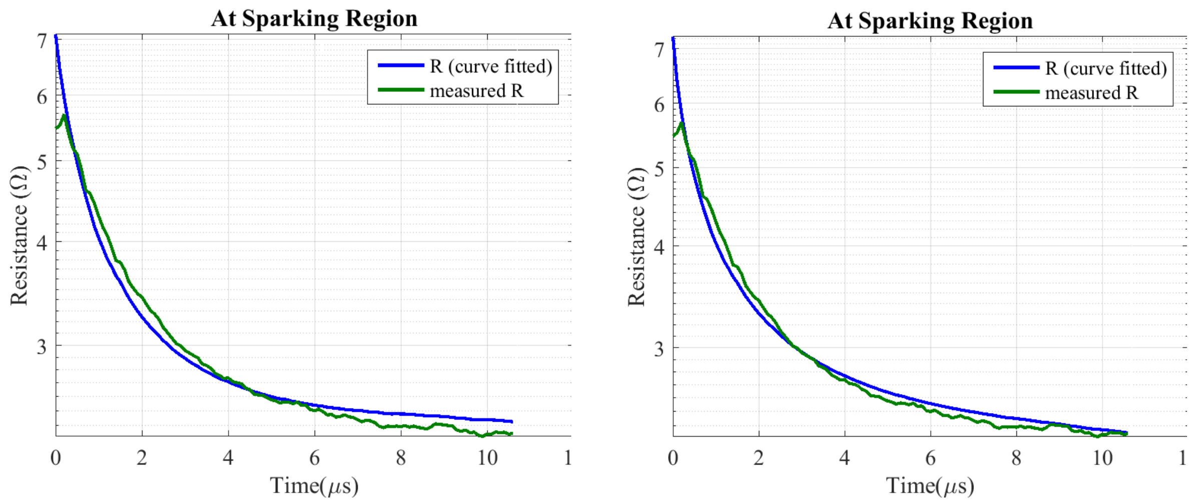
- Plots for experimental voltage, measured R and current at sparking region are shown in Fig.5



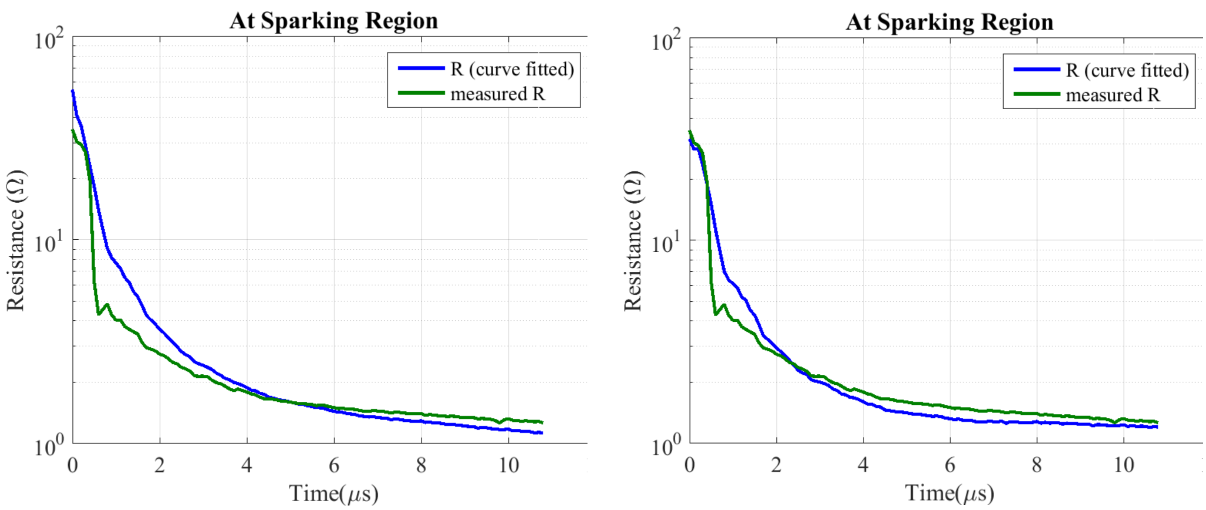
a) Silicon b) Steel

Fig: 5

- Plots for Sparking resistance models proposed in equation 1 & equation 10 are shown in Fig 6.



a) Silicon: curve fitting using current b) Silicon: curve fitting using current and time



c) **S**teel: curve fitting using current d) **S**teel: curve fitting using current and time

Fig: 6

- Adjusted R-squared for steel and silicon is plotted for both the cases as shown in Fig 7.

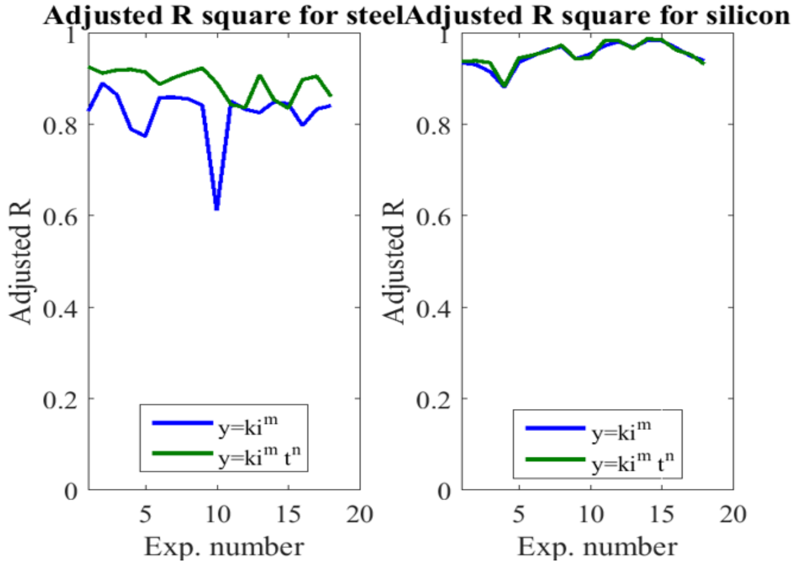


Fig 7: Adjusted R squared for steel and silicon

3.2 **Slicing rate**

- Number of neurons required in 1st hidden layer is decided experimentally by calculating square error for each case as shown in Fig 8(a). [5] The model which has less error is selected. Values of dependent variable are divided by maximum value of dependent variable so that it can be in range of 0 to 1 as activation function used in hidden layer is *tanh* which gives output in range of [-1 ,1]. By doing this values of dependent variable are in [-1,1].
- Square error is calculated between output of neural network and values of dependent variable after scaling in range [0,1].

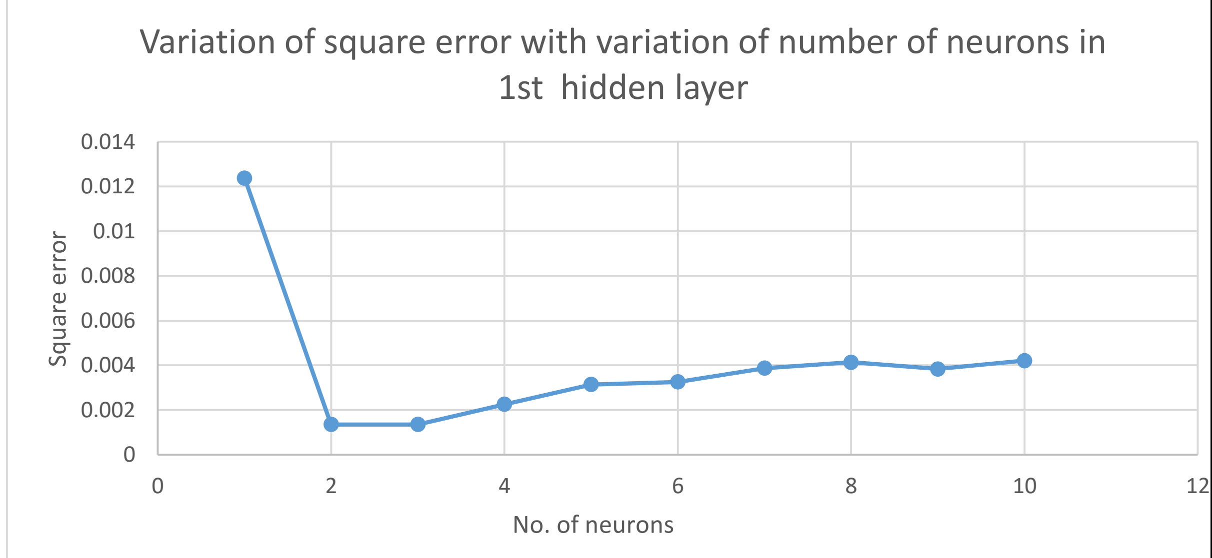


Fig 8(a): Variation of square error in ANN with one hidden layer for slicing rate

- For optimum value of neurons in 1st hidden layer, number of neurons in second layer are decided by same procedure i.e. using square error method as shown in Fig (8b).
- Adding more layers increasing square error which implies overfitting of the training set. So, only one hidden layer is selected. [4]

Table 5: SE in ANN with two hidden layers for slicing rate

S. No.	No. of neurons in 1st hidden layer	No. of neurons in 2nd hidden layer	Square Error		Average
			1	2	0.004566
1	2	1	0.004183	0.004949	0.00323
2		2	0.00317	0.003291	0.005152
3		3	0.004935	0.00537	0.007056
4		4	0.006659	0.007453	0.005057
5		5	0.004806	0.005307	0.005962
6		6	0.005376	0.006548	0.005845
7		7	0.00616	0.00553	0.005874
8		8	0.006171	0.005577	0.00585
9		9	0.006212	0.005489	0.005295
10		10	0.005409	0.005181	0.004566
11	3	1	0.0067	0.003611	0.003539
12		2	0.004199	0.004682	0.00444
13		3	0.005801	0.005786	0.005794
14		4	0.006666	0.007008	0.006837
15		5	0.005801	0.006967	0.006384
16		6	0.005773	0.005974	0.005874
17		7	0.00592	0.005967	0.005943
18		8	0.006307	0.005844	0.006076
19		9	0.005835	0.006803	0.006319
20		10	0.005316	0.004504	0.00491

- From Table 5 data is plotted and shown in Fig (8b).

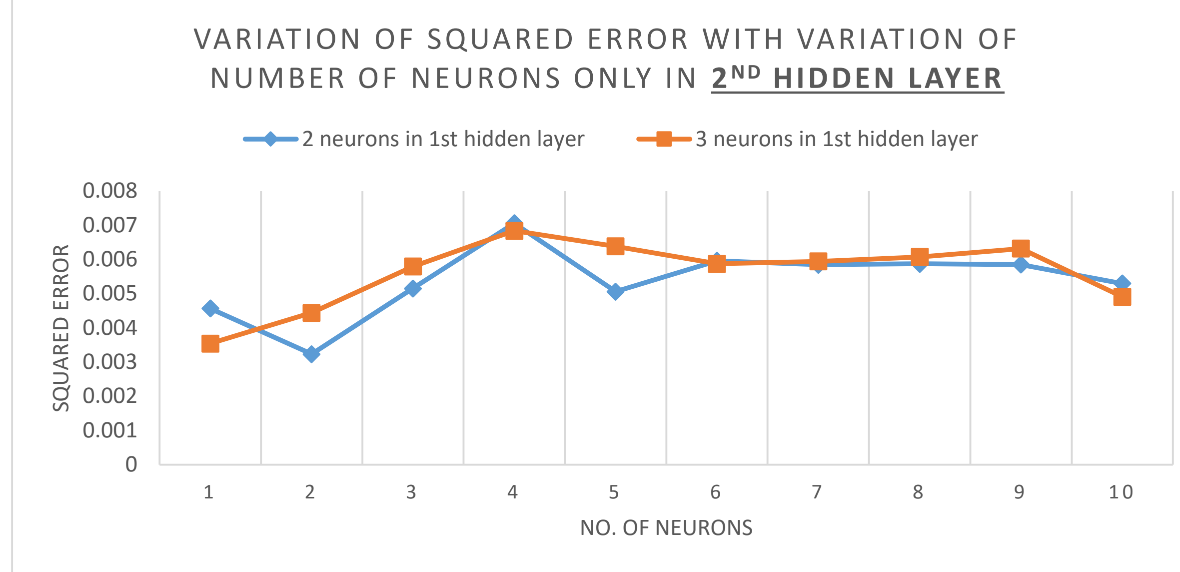
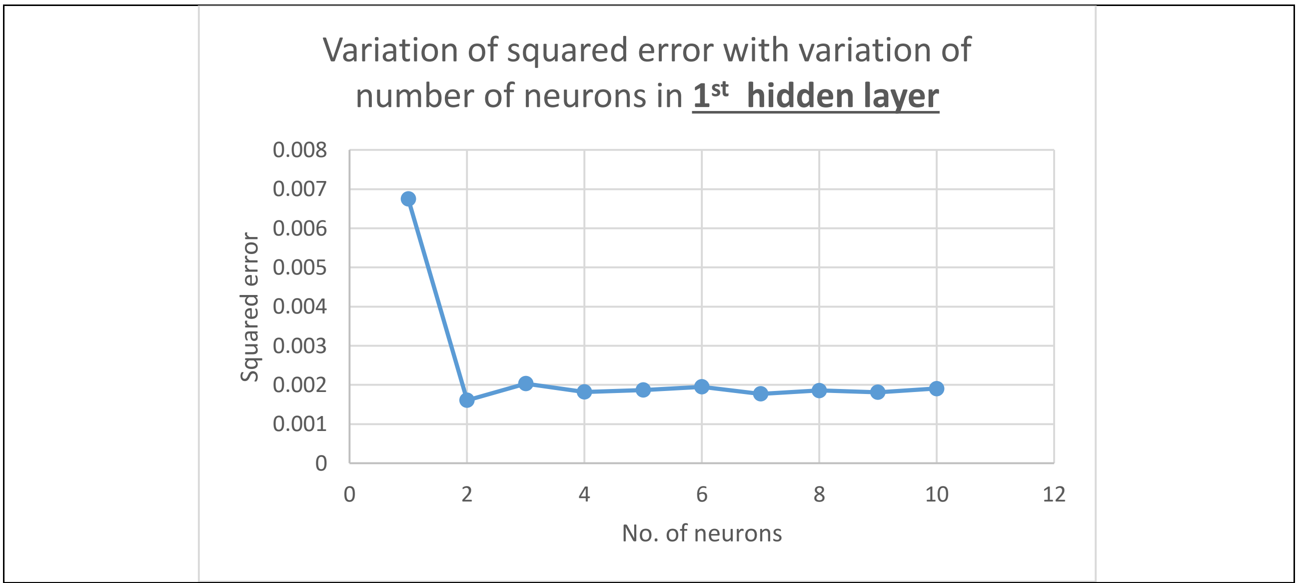


Fig (8b): Variation of squared error in ANN with two hidden layers for slicing rate

- It can be observed from the above two figures that square error for two neurons in hidden layer 1 and zero neurons in hidden layer 2, is minimum.
- Squared error in case of multiple linear regression (MLR) = 0.002222 and square error in case of ANN = 0.001331.

3.3 Kerf loss

Fig 9 shows the variation squared error in ANN for kerf loss. It is the amount of material lost in the process of cutting. How it changes by changing input parameters is shown in Table 2.



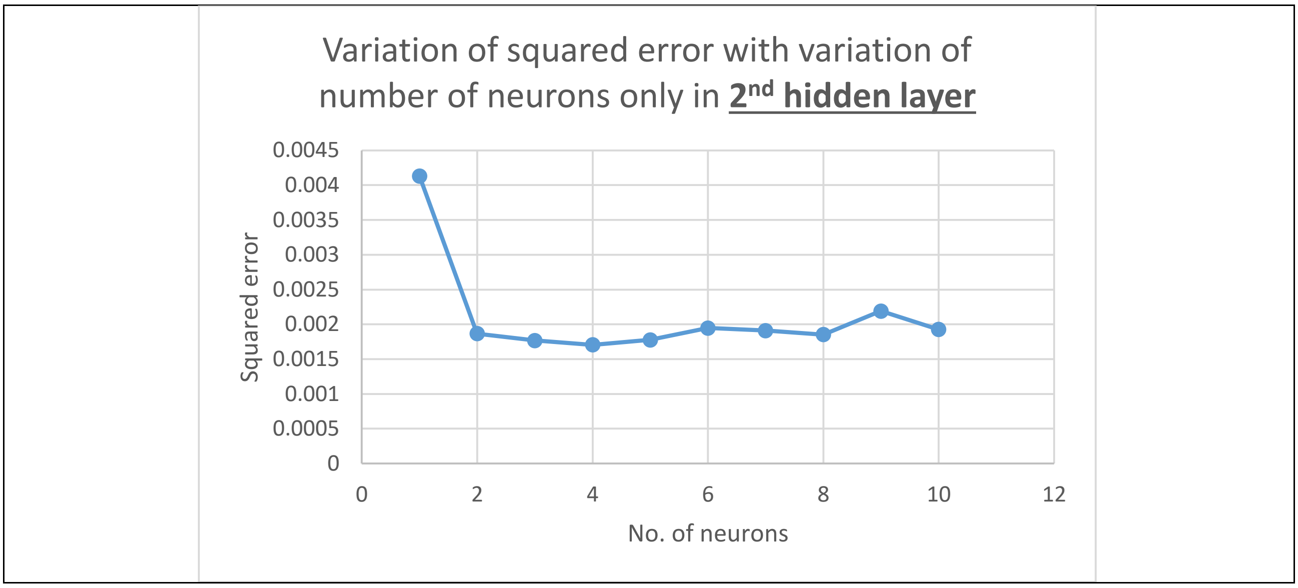
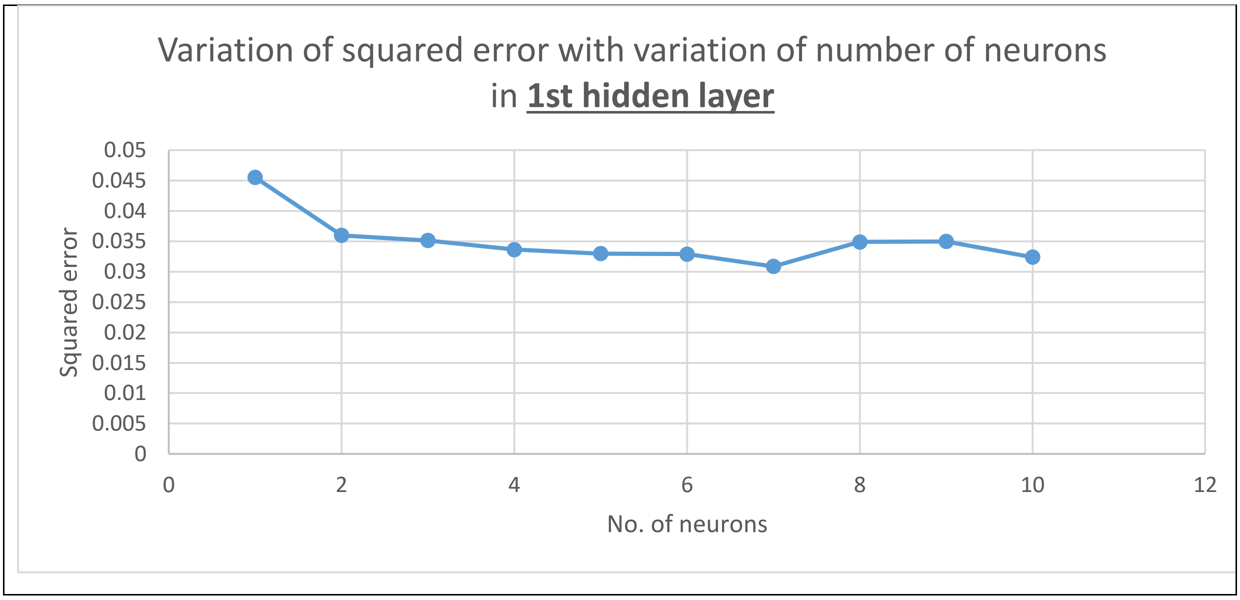


Fig 9: Variation of squared error in ANN for kerf loss

- It can be observed from the figure 9 that squared error for two neurons in hidden layer 1 and zero neurons in hidden layer 2, is minimum.
- Squared error in case of multiple linear regression (MLR) = 0.001881 and squared error in case of ANN = 0.001609.

3.4 Thermal damage

Fig 10 shows the variation squared error in ANN for thermal damage. How it changes by changing input parameters is shown in Table 2.



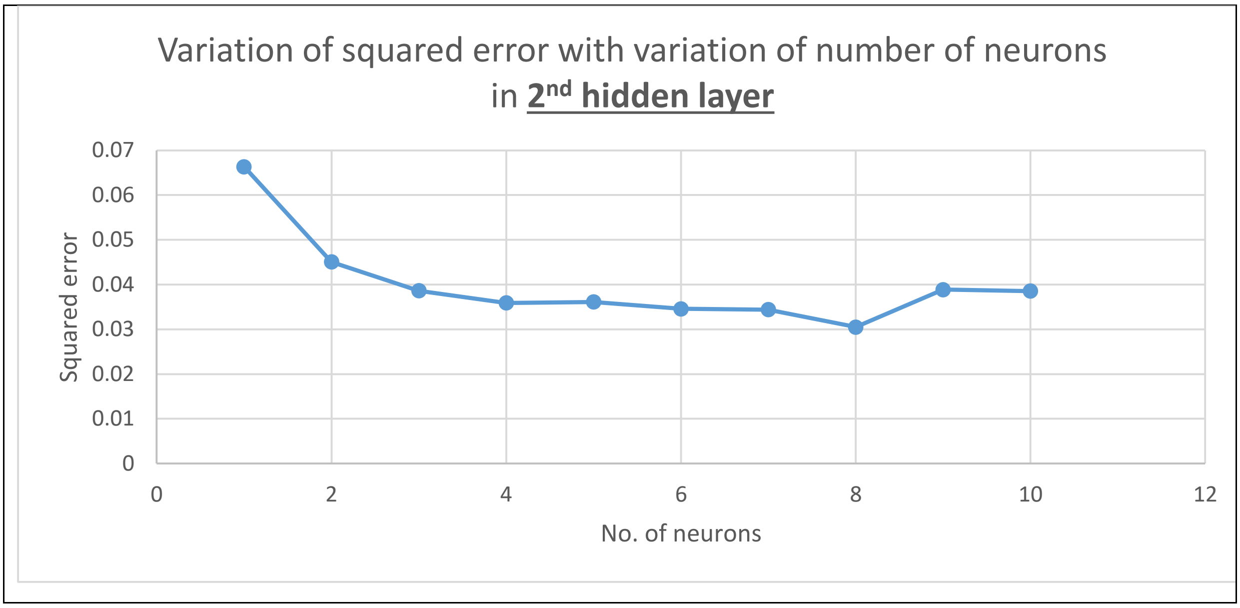


Fig 10: Variation of squared error in ANN for thermal damage

- It can be observed from the Fig 10 figure that squared error for seven neurons in hidden layer 1 and eight neurons in hidden layer 2, is minimum.
- Squared error in case of multiple linear regression (MLR) = 0. 0.009565 and squared error in case of ANN = 0. 0.008933.

4 Conclusion

Spark resistance

- Voltage pulses of silicon and steel WEDM are not similar. Steel voltage pulse has some time delay. So, semiconductors WEDM should be studied separately.
- Two models are proposed for spark resistance.
  1 $\boldsymbol{R = ki^m}$
  2 $\boldsymbol{R = ki^m t^n}$
  Adjusted R-squared is used to compare models having different number of parameters [2]. Here, curve fitting using model 2 in case of steel is better than model 1.
- Adjusted R-squared is not improved in case of Silicon but shows a significant improvement in case of steel. This implies that spark resistance is more dependent on time in case of steel than silicon.

Neural network

- Squared error in neural network is less than multiple linear regression(MLR) which implies that neural network predicts kerf loss, slicing rate and thermal damage better than MLR.

- Neural network modelling has been done as proper equations for finding kerf-loss, thermal damage and slicing rate are not available and also the WEDM process is stochastic in nature.
- Neural network can predict kerf-loss, thermal damage and slicing rate with reasonable accuracy than MLR.
- Neural network model can be used as a functional block in control model of WEDM which will give slicing rate, kerf-loss and thermal damage.

References

1. R. Montano, M. Becerra, V. Cooray, M. Rahman, P. Liyanage, 2006, Resistance of Spark Channels, IEEE Transactions on Plasma Science, vol. 34, no. 5, pp. 1610-1619
2. Regression Analysis by Example, Fourth Edition, WILEY SERIES IN PROBABILITY AND STATISTICS established by WALTER A. SHEWHART and SAMUEL S. WILKS.
3. T.G. Engel, A.L. Donaldson, M. Kristiansen, 1989, The pulsed discharge arc resistance and its functional behavior, IEEE Transactions on Plasma Science, vol. 17, no. 2, pp. 323-329.
4. Coursera machine learning course by Andrew Ng (Week 4 and 5)
5. S. Assarzadeh, M. Ghoreishi, 2007, Neural-network-based modeling and optimization of the electro-discharge machining process, The International Journal of Advanced Manufacturing Technology, vol. 39, no. 5-6, pp. 488-500
6. “Wire EDM | engineering clicks” , internet: http://www.emgineeringclicks.com/wire-edm/ , 11-07-2019.
7. “Machine learning - Discussion|coursera” ,internet:https://www.coursera.org/learn/machine-learning/ ,11-07-2019.