INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 4, Issue 6)
Available online at: www.ijariit.com

Forming a mathematical relationship between voltage generated and sound inputted using piezoelectric plates

Ritvik Singh
mailtosinghritvik@gmail.com
Army Public School, Shillong, Meghalaya

ABSTRACT

Today's rate of population growth is soaring higher than ever and shows no signs of regressing. Consequently, more and more resources are being consumed at an accelerated rate. In lieu of this energy and resource crisis, many alternate energy harvesting systems are being designed. This project aims to understand and appreciate the potential of sound via piezoelectric materials as a source of electricity. The aim is to recognize the relation between frequency and SPL of a sound wave with the voltage that the sound is capable of producing.

Keywords— *Piezoelectricity, Neural Networks, Back-propagation, ANN, Alternate energy*

1. INTRODUCTION

1.1 Piezoelectricity

The direct piezoelectric effect is that these materials, when subjected to mechanical stress, generate an electric charge proportional to that stress. The inverse effect piezoelectric effect is that these materials become strained when an electric field is applied, the strain again being proportional to the applied field. Clever use of piezoelectric materials enables the realization of a wide variety of technical functions.[1]

The piezoelectric effect results from the linear electromechanical interaction between the mechanical and electrical states in crystalline materials with no inversion symmetry. For example, lead zirconate titanate crystals will generate measurable piezoelectricity when their static structure is deformed by about 0.1% of the original dimension. Conversely, those same crystals will change about 0.1% of their static dimension when an external electric field is applied to the material. [2]

$$V = h\Delta x/d \tag{1}$$

The above-mentioned equation effectively relates voltage V to the transverse displacement Δx.
Here, h is the height of the ceramic layer and d is the deformation constant. This gives us equation (1)

1.2 Neural Network and Machine learning

Machine learning is a subset of artificial intelligence in the field of computer science that often uses statistical techniques to give computers the ability to learn (gradually improve performance on a task) with data, without being explicitly programmed.

Machine learning tends to be useful when the user cannot determine the proper relation between his input variables and output, but he has enough experimental data to understand and recognize a pattern.

An Artificial Neural Network (ANN) is a mathematical model that tries to simulate the structure and functionalities of biological neural networks. The basic building block of every artificial neural network is an artificial neuron, that is, a simple mathematical model (function). Such a model has three simple sets of rules: multiplication, summation, and activation.

At the entrance of artificial, neuron the inputs are weighted what means that every input value is multiplied with individual weight. In the middle section of an artificial neuron is sum function that sums all weighted inputs and bias. At the exit of an artificial neuron, the sum of previously weighted inputs and bias is passing through an activation function that is also called transfer function.[3](More on this in the next section, along with back propagation)

1.3 Human Sound

An audio filled with loud crowd noise is used and the frequency distribution is analyzed. The loudness of the sound at a given point is measured in decibels, which can be calculated as follows

$$SPL = 10log\frac{P}{P_0} \tag{2}$$

Here, P is the current pressure, and P_0 is the reference pressure

As evident, this is a logarithmic function. The analysis shows that the noise of a crowd of people can range from 110 dB to higher 80 dB. Thus all our experiments are conducted at amplitudes about this range. Apart from this, the frequency of the sound ranges from 100 Hz to 800 Hz, which is approximately the frequency which the human crowd emits.

2. FORMULAE

To create the required relation, we use:

$$SPL = 10log\frac{P}{P_0}\text{ dB} \tag{3}$$

$$V = h\Delta x/d \tag{4}$$

$$I = \frac{\Delta P^2}{2\varrho v} \tag{5}$$

$$\boldsymbol{I = 2\pi^2 \varrho f^2 v \Delta x^2} \tag{6}$$

Here, $\boldsymbol{I}$ is the intensity of sound, ϱ is the density of medium (air), $\boldsymbol{f}$ is the frequency of the sound, and $\boldsymbol{v}$ is the velocity.

Apart from this, P_0 is taken as 20 μPa in air. Thus,
Using the above relations we get,

$$V = (h\Delta p/2\varrho v f \pi d) \tag{7}$$

$$\boldsymbol{V} = \frac{hP_0(10^{SPL/10} - 1)}{2\varrho v f d\pi} \tag{8}$$

The rest of the paper aims to verify this relation.

3. MATERIALS

(1) Piezoelectric transducers (Here, EEPiezo25mm)
(2) Decibel Meter (Preferably PCE-MSL 1)
(3) Frequency Meter (Any)
(4) Speaker (Powerful)
(5) Multimeter

4. SOFTWARE

(1) Audacity/ any other audio editor
(2) Simbrain/ any other ANN implementation software.

5. EXPERIMENT AND DATA

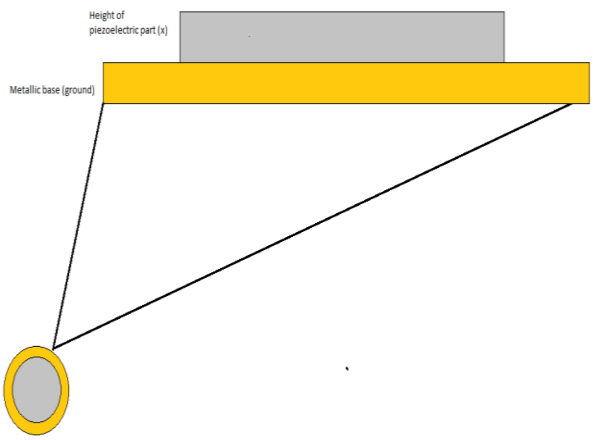

Fig. 1: Side view diagram of a piezoelectric transducer plate

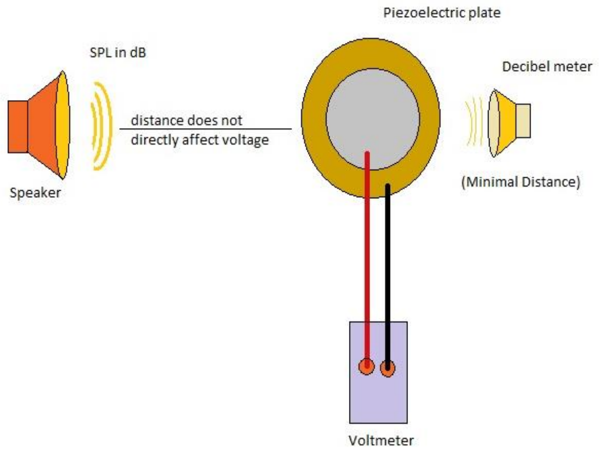

Fig. 2: Diagrammatic representation of the experimental setup

As we see, the piezoelectric plate we used has 0.25mm thick piezoelectric material, the rest is a conducting metal that can be used to transfer the potential, and act as the ground. The piezoelectric plate is placed perpendicularly to the source speaker to avoid any variation in sound. The piezoelectric transducer is connected to a multimeter to monitor AC voltage. The Decibel meter is placed next to the plate (as close as possible) to measure the SPL at that point. Using the software, we emit sine/square and triangle waves of the following frequencies. (The frequency meter is used to validate the frequency).

A.C. (~) voltage is monitored and recorded. The following data is generated.

Table 1: Dataset containing the experimental values

Amplitude (dB)	Frequency (Hz)	Voltage (V)
106	**811**	**0.291**
106	805	0.295
106	800	0.298
106	795	0.3
106	791	0.302
99	811	0.234
99	805	0.238
99	800	0.24
99	795	0.243
99	791	0.25
96	811	0.194
96	805	0.199
96	800	0.208
96	795	0.212
96	791	0.221
92	811	0.169
92	805	0.174
92	800	0.176
92	795	0.178
92	791	0.181
106	711	0.329
106	705	0.361
106	700	0.381
106	695	0.402
106	691	0.421
99	711	0.301
99	705	0.329
99	700	0.345
99	695	0.366
99	691	0.381
96	711	0.273
96	705	0.286
96	700	0.292
96	695	0.3
96	691	0.311
92	711	0.241
92	705	0.249
92	700	0.252
92	695	0.256
92	691	0.261
106	611	0.505
106	605	0.539
106	600	0.549
106	595	0.562
106	591	0.591
99	611	0.497
99	605	0.518
99	600	0.523
99	595	0.528
99	591	0.535
96	611	0.472
96	605	0.482
96	600	0.488
96	595	0.493
96	591	0.5
92	611	0.308
92	605	0.341

92	600	0.362
92	595	0.384
92	591	0.413
106	511	0.902
106	505	0.946
106	500	0.962
106	495	0.984
106	491	1.001
99	511	0.771
99	505	0.8
99	500	0.823
99	495	0.846
99	491	0.862
96	511	0.722
96	505	0.74
96	500	0.751
96	495	0.762
96	491	0.774
92	511	0.688
92	505	0.698
92	500	0.706
92	495	0.711
92	491	0.716
106	411	0.924
106	405	0.966
106	400	0.982
106	395	1.002
106	391	1.017
99	411	0.887
99	405	0.9
99	400	0.921
99	395	0.942
99	391	0.956
96	411	0.87
96	405	0.882
96	400	0.889
96	395	0.901
96	391	0.905
92	411	0.845
92	405	0.862
92	400	0.878
92	395	0.884
92	391	0.892
106	311	1.096
106	305	1.102
106	300	1.106
106	295	1.116
106	291	1.12
99	311	1
99	305	1.019
99	300	1.033
99	295	1.054
99	291	1.072
96	311	0.997
96	305	1.014
96	300	1.038
96	295	1.054
96	291	1.063
92	311	0.944
92	305	0.958
92	300	0.964
92	295	0.975
92	291	0.99
90	211	0.139

90	205	0.143
90	200	0.145
90	195	0.146
90	191	0.147
86	211	0.132
86	205	0.135
86	200	0.138
86	195	0.139
86	191	0.141
84	211	0.087
84	205	0.09
84	200	0.091
84	195	0.092
84	191	0.094
82	211	0.062
82	205	0.066
82	200	0.068
82	195	0.071
82	191	0.072
83	111	0.072
83	105	0.072
83	100	0.073
83	95	0.073
83	91	0.074
81	111	0.051
81	105	0.052
81	100	0.052
81	95	0.054
81	91	0.057
77	111	0.021
77	105	0.023
77	100	0.024
77	95	0.026
77	91	0.029

6. OBSERVATION

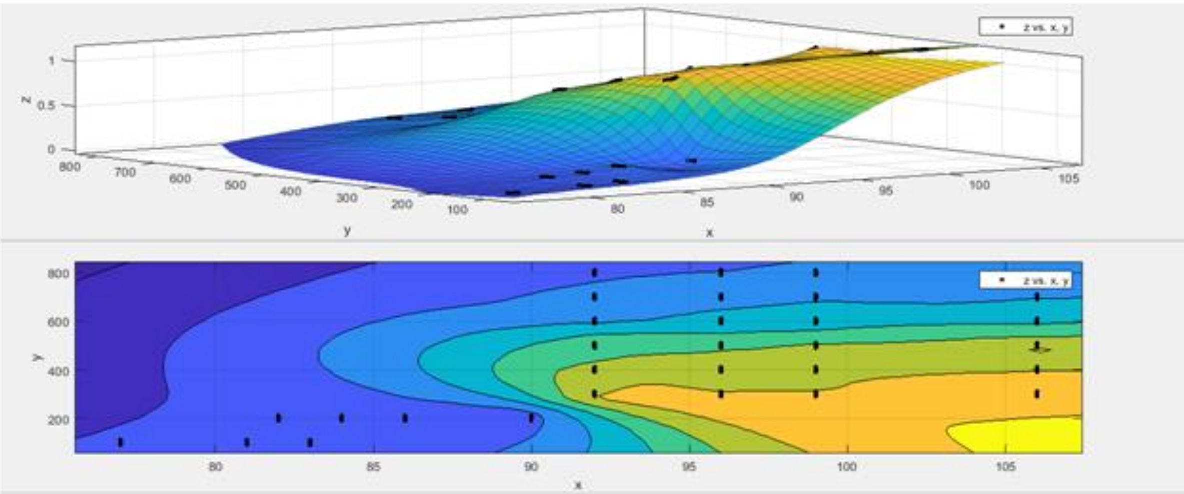

Fig. 3: Graph illustrating the aforementioned dataset

The upper trace shows all the variable marking the peak voltage generated by the piezoelectric plate (yellow region). The lower trace is the contour showing the region of the voltage generated with db and frequency on sleeping (x-axis) and standing (Y-axis) respectively. With these graphs, one can make out (or explain) the dependence of generated voltage on SPL and frequency of sound from the source.

It is observed that:

$$V\ \alpha\ \frac{1}{f^{x}} \text{ and } V\alpha\ y^{SPL} \text{ (for some } x, \text{y)} \tag{9}$$

This is in agreement to equation (8).
By comparison, we can observe that x = 1 and y = 10
Except for d, the deformation constant, all the other values are known. One may find ‘d’ using the experimental data, and plugging in the values in equation (8)

The mean value collected from the dataset gives d = 7.34139e-06
And standard deviation is = 1.99452e-06.

7. NEURAL NETWORK ANALYSIS

The backpropogation algorithm trains a given feed-forward multilayer neural network for a given set of input patterns with known classifications. When each entry of the sample set is presented to the network, the network examines its output response to the sample input pattern. The output response is then compared to the known and desired output and the error value is calculated. Based on the error, the connection weights are adjusted. The backpropogation algorithm is based on *Widrow-Hoff delta learning rule* in which the weight adjustment is done through the *mean square error* of the output response to the input. The set of these sample patterns are repeatedly presented to the network until the error value is minimized.

To effectively use gradient descent or stochastic gradient descent, we need an efficient way of computing the gradient of an error function. Backpropogation provides such a way.

This network is initialized and configured in the following way.

(1) The inputs to be given to the neural network are determined, along with the number of hidden layers and the neurons in each hidden layer
(2) The weights and biases are initialized for each layer From the input layer, the input vectors (I_i) are multiplied with the initialized weight ($w_{i,1}$) and summed with the respective node bias ($b_{i,1}$) given by

$$a_{1,1} = \sum_{i=1}^{n} (I_i w_{i,1}) + b_{i,1} \tag{10}$$

(3) This sum is given to an activation function which is often a sigmoid (log or tan). The most commonly used activation function is log sigmoid (σ) given by

$$\sigma(a_{1,1}) = \frac{1}{1 + e^{-a_{1,1}}} \tag{11}$$

(4) The output of the activation function acts as the input to the next layer and step 3–4 are repeated until the output layer is reached.
(5) After reaching the output layer, the value of the output parameter obtained is examined and compared with the target parameter.
(6) Performance parameters like MSE or SSE along with gradient are calculated as per the training algorithm specifications.

$$MSE = (\frac{1}{n} \sum_{i=1}^{n} \sum_{i=1}^{k} (t_{i,k} - o_{i,k})^2 \tag{12}$$

Where $t_{i,k}$ is the target value and $o_{i,k}$is the output obtained.[7] We can represent Inputs, Weights, and Outputs using matrices.[8]

In the matrix representing the input, hidden and output nodes are represented by three vectors i, h and o respectively. The weights connecting each layer are represented by a matrix. V connects the input layer with the hidden layer, and W connects the hidden layer with the output layer. For simplicity, we’re showing one such representation using the following figure.

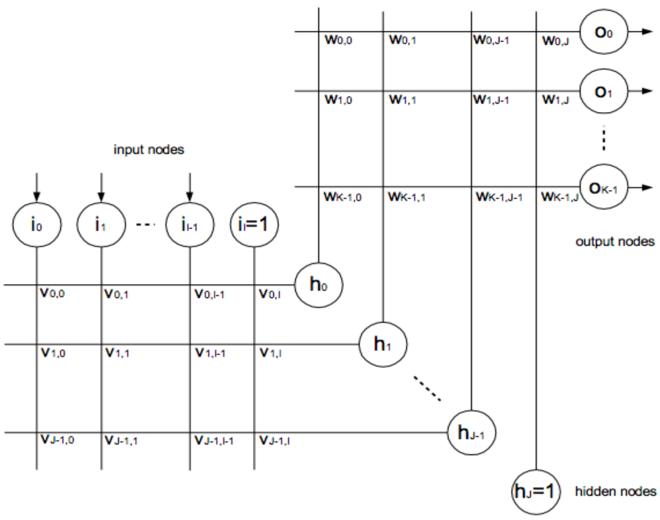

Fig. 4: Matrix representation

In our case, we have 2 input nodes - frequency and SPL, one output node - voltage. The number of hidden layers has been set experimentally. All of the following Backpropogation ANNs have been created and trained using Simbrain.jar

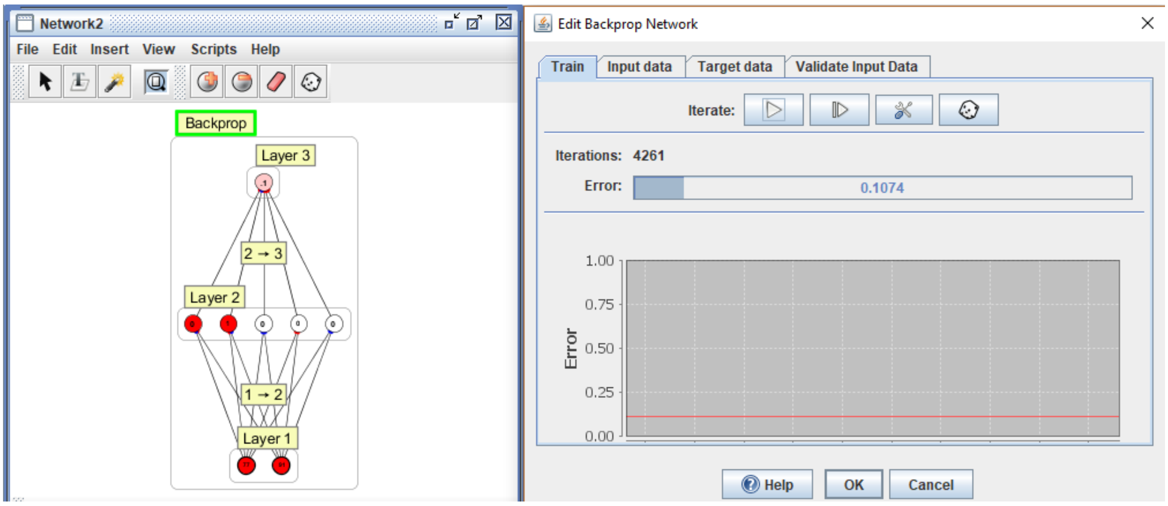

Fig. 5: One layer ANN

In the above Backprop ANN, we have one hidden layer containing five neurons. In the dialogue box on the right, we see that the error is 0.1074 after 780 randomizations and 4261 iterations.

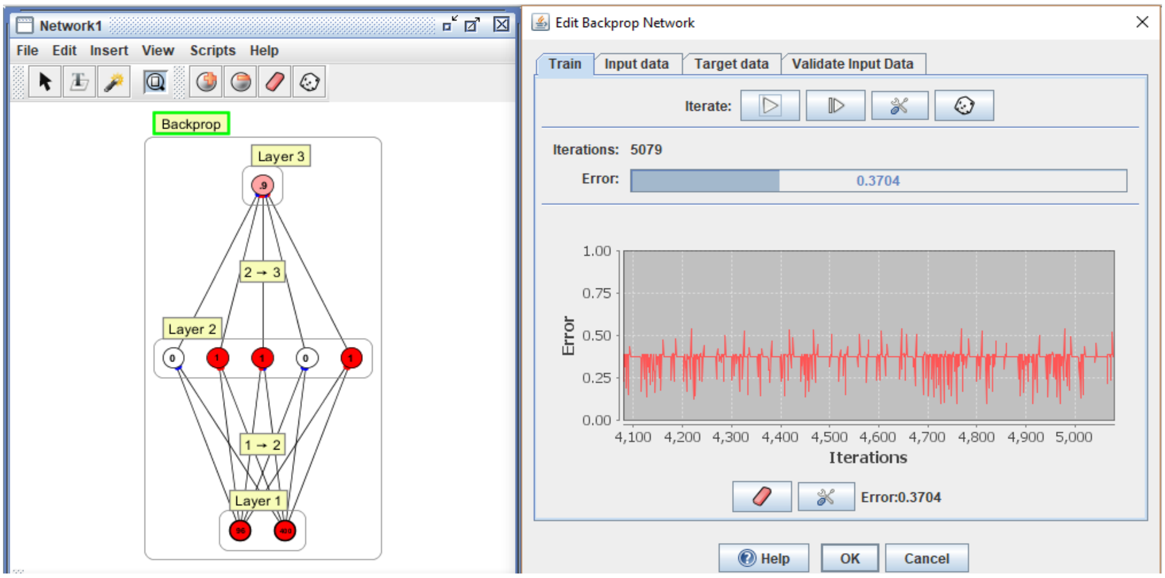

Fig. 6: One layer ANN with error

However, after some more iterations and randomizations, the model begins to fluctuate in the error calculation. Thus, it is discarded.

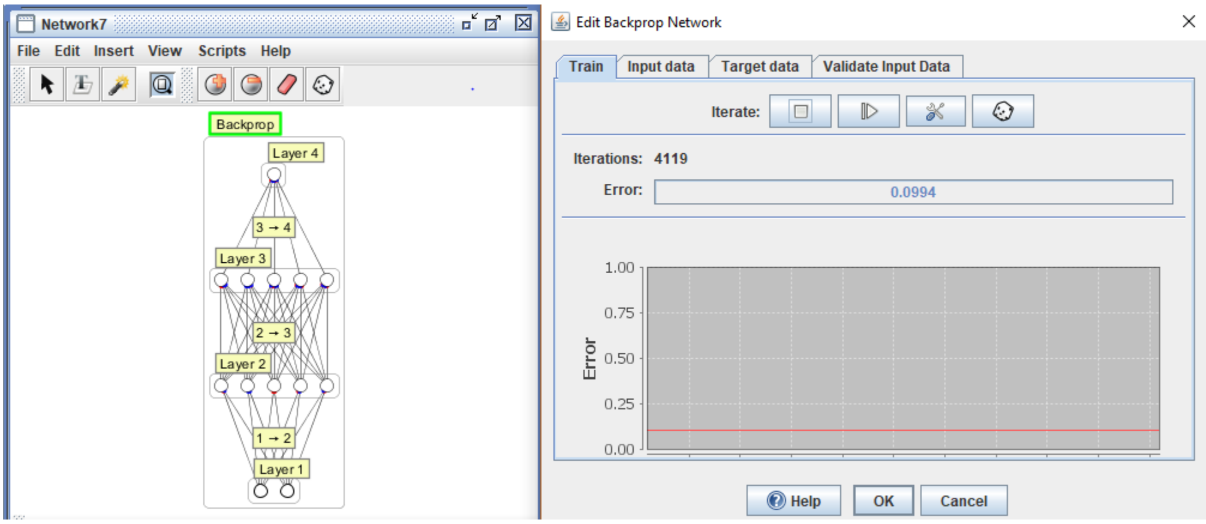

Fig. 7: Two-layer ANN

In the above Backprop ANN, we have two hidden layers containing five neurons. In the dialogue box on the right, we see that the error is 0.0994 after 1023 randomizations and 4119 iterations. However, this error soon fluctuated after some more training. Thus, we rejected this model.

In the following Backprop ANN, we have 3 hidden layers containing five neurons. In the dialogue box on the right, we see that the error is 0.1181 after 780 randomizations and 3671 iterations. Once again, despite having more number of layers, the error is high. Thus, we discarded this model too.

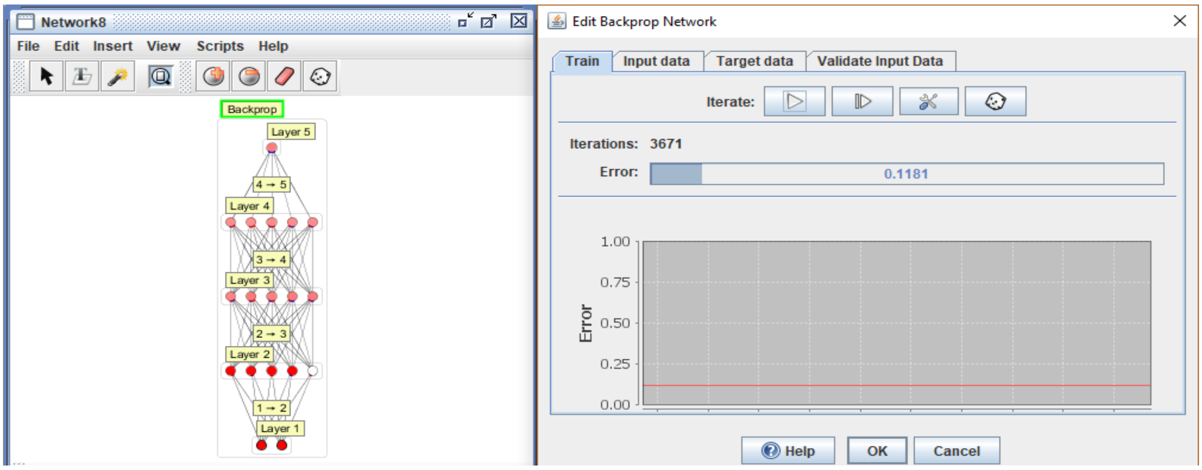

Fig. 8: Three layer ANN

In the following Backprop ANN, we have 4 hidden layers containing five neurons. In the dialogue box on the right, we see that the error is 0.1181 after 5641 iterations and 780 randomisations. Once again, despite having more number of layers, the error is high. Thus, we discarded this model too.

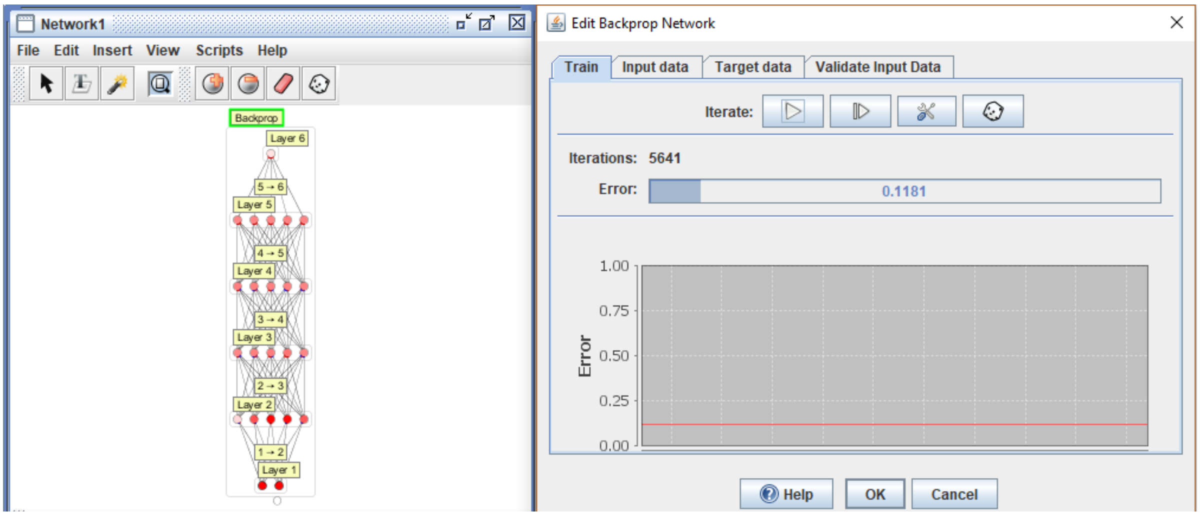

Fig. 9: Four layer ANN

The following Backprop ANN, possessing five hidden layers, has an error equal to 0.0994 after 5857 iterations and 785 randomizations. Thus, it manages the lowest error and remains stable too. However, as we see, a practical application of this model can be very inefficient

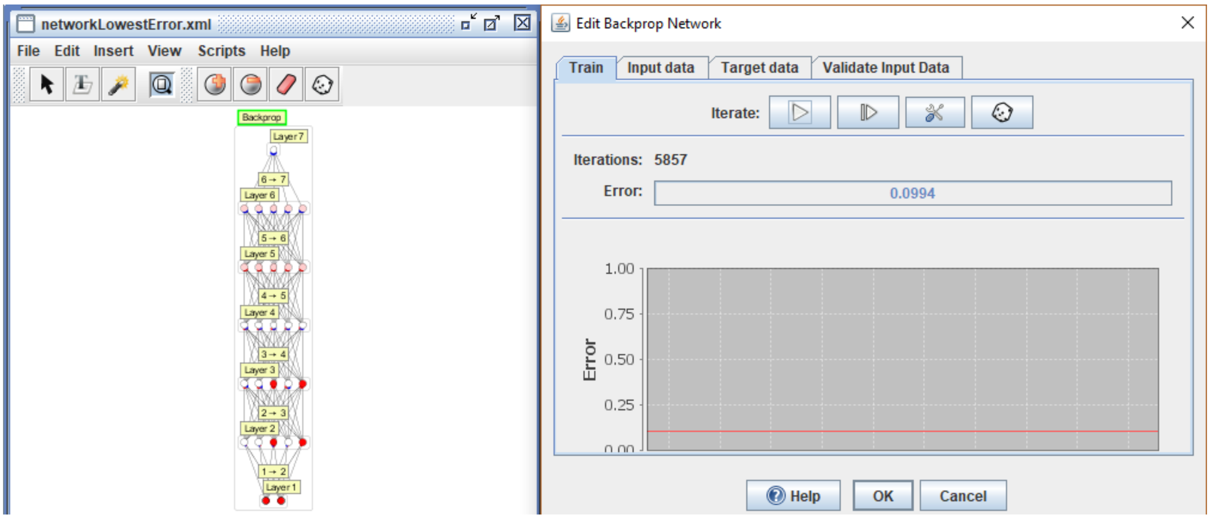

Fig. 10: Five layer ANN

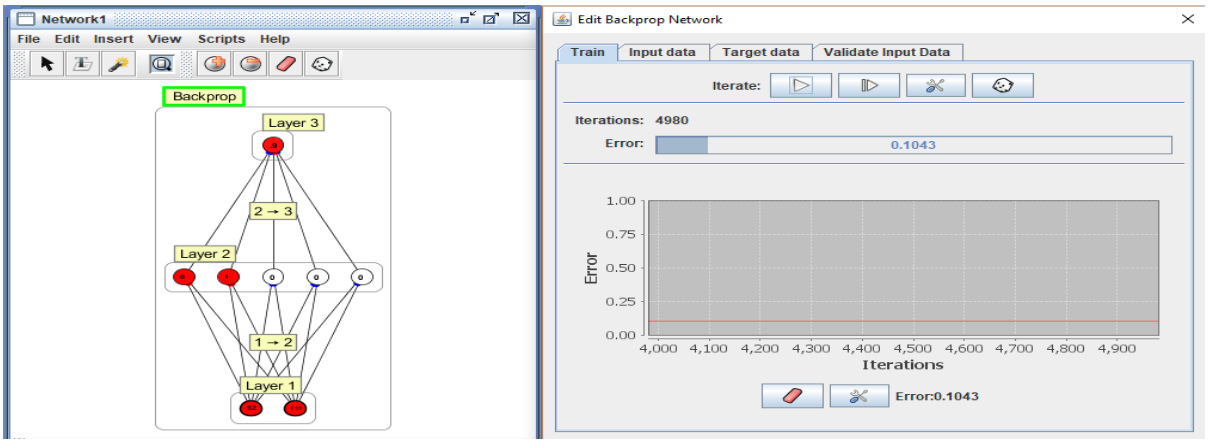

Fig. 11: One layer ANN (revisited)

Another model possessing only one hidden layer manages to bring the error down to 0.1043 after 4980 iterations and 452 randomizations. Unlike the first neural network, the learning rate has been decreased to 0.20 from 0.25, and momentum is 0.8. This is the most effective and efficient model. Hence, we shall use this model for the rest of the work, and future development.

Table 2: Features of the various ANN

No. of layers	Errors	Activation Function	Nature	Learning Rate	Momentum
1	0.1074	Sigmoid	Fluctuating	0.25	0.9
2	0.0994	Sigmoid	Fluctuating	0.25	0.9
3	0.1181	Sigmoid	Stable	0.25	0.9
4	0.1181	Sigmoid	Stable	0.25	0.9
5	0.0994	Sigmoid	Stable	0.25	0.9
1	0.1072	TanH	Unstable	0.25	0.9
1	0.1043	Sigmoid	Stable	0.20	0.8

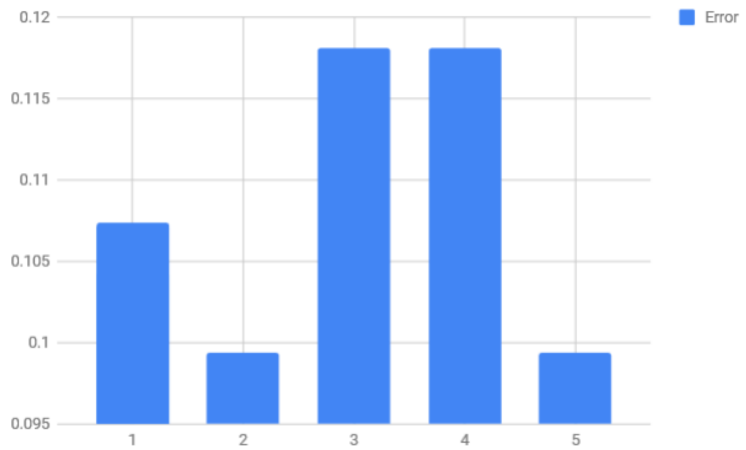

Fig. 12: Error at 0.25 learning rate and 0.9 momenta with the Sigmoid activation function

Using a C++ program (in Turbo C++), we find out the standard deviation between the voltage from the dataset and the voltage from the equation.

Comparison between the output from dataset and that observed using the equation shows that standard deviation is = 1.9952e-02.

As we see, this error is very less and can be accounted for by experimental or personal errors.

8. CONCLUSION

- The given voltage generated by a piezoelectric plate:
- Proportional to P_o raised to SPL of the sound
- Is inversely proportional to the frequency.
- Directly proportional to the height of the piezoelectric element layers
- Inversely proportional to the density
- Inversely proportional to the velocity of the sound
- To be precise:

$$\boldsymbol{V} = \frac{hP_0(10^{SPL/10} - 1)}{2\varrho v f d\pi}$$

9. REFERENCES

[1] An Introduction to Piezoelectric Materials and Applications by Jan Holterman and Pim Groen (1.2 Piezoelectricity, Introduction)
[2] Advance in Electronic and Electric Engineering. ISSN 2231-1297, Volume 4, Number 3 (2014), pp. 313-318 © Research India Publications
[3] Andrej Krenker, Janez Bešter and Andrej Kos (2011). Introduction to the Artificial Neural Networks, Artificial Neural Networks - Methodological Advances and Biomedical Applications, Prof. Kenji Suzuki (Ed.), ISBN: 978- 953-307-243-2, InTech.
[4] http://www.rfwireless-world.com/Terminology/Ferroelectric-vs-piezoelectric.html
[5] Identification of piezoelectric hysteresis by a novel Duhem model based neural network Geng Wang∗, Guoqiang Chen School of Mechanical and Power Engineering, Henan Polytechnic University, Jiaozuo, 454003, China
[6] *Kiyosh Kawaguchi 2000-06-17* http://wwwold.ece.utep.edu/research/webfuzzy/docs/kk-thesis/kk-thesis-html/node22.html
[7] Power prediction of soiled PV module with neural networks using hybrid data clustering and division techniques Subrahmanyam Pulipaka, Rajneesh Kumar Department of Electrical and Electronics Engineering, Birla Institute of Technology and Science, Pilani 333031, India
[8] Christensen, B. K. (2003). Matrix representation of a Neural Network. Technical University of Denmark (DTU).

