Test for Efficient Increase/ Decrease Factors for Resilient Backpropagation using Combustion Engine Vibration Signals

N. D. Liyanagedera, A. Ratnaweera, D. I. B. Randeniya

Abstract* - *Resilient backpropagation is a recently emerging neural network with a high potential. This neural network is capable of handling a network structure with a large number of input nodes which many other networks fail. This is achieved based on the way the weights are updated in the algorithm. Resilient backpropagation uses two constant values, decrease factor [η^-] and increase factor [η^+] to update the weights and to get the optimal solution. This experiment checks on to find how the neural network performs when different values were used for these constants. The training, testing and validation of the neural networks were done using vibration signal data collected from a combustion engine corresponding to 16 different fault combinations available in the combustion engine. The performances of the networks were compared using mean square error, time and epoch. The final results indicate that, when the decrease factor is in the range of 0.5 to 0.6 and when the increase factor is in the range of 1.2 to 1.3 the resilient backpropagation algorithm has the best performance.

Index terms* – *Resilient Backpropagation Algorithm, Increase/Decrease Factor, Combustion Engine, Vibration Signals.

I. INTRODUCTION

Machinery fault diagnosis [1] using collected data from a machine is currently an active research area. Fault detection of an internal combustion engine [2] is one such instance and data collected from engine oil analysis [3], exhaust gas analysis and vibration signal analysis [4] can be used in such experiments. Vibration signals collected from a combustion engine [Figure 1] was used as the data for this research. The collected vibration signals were modified according to the needs of the research and Fourier transformation [Figure 2] was one such modification. The relationship between data collected from a mechanical device and the characteristics of that device is very complex and non linear. To map such a highly non linear relationship, the use of expert systems like neural networks can be of high value [5] as neural networks have the ability to map two data sets having a complex non linear relationship. [6]. Few researches have been done to detect faults in a combustion engine, but such researches are limited to a small area because of the complexity of the data. The recently published paper [7] talks about several faults that could be available in a combustion engine, but the neural network detects a single fault when only that fault is available in the machine. This has major practical limitations because, for a 4 fault environment 16 possible fault combinations are available and assuming, at a given instance the engine can have only one fault will highly reduce the usefulness of the developed system. While improving such weakness, this research concentrates on studying several faults in a combustion engine and detecting any fault combination among the selected faults.

This experiment was conducted using vibration signals collected from a combustion engine. The results shown in this paper is related with the work done and published in the past. The following part briefly explains the past work as it is important for the results in this paper as well. A non linear relationship between an internal combustion engine and vibration signals obtained from the engine [Figure 3] was assumed. This relationship was proved with the aid of a neural network model. [8] The relationship was tested for four faults in the combustion engine and they were bearing, fan imbalance, parallel misalignment and angular misalignment. Vibration signals were obtained for all 16 types of fault combinations such as, for no fault condition and for all instances of 1 error, 2 errors, 3 errors and 4 errors situations. The neural network training [Figure 4], testing and validation was done for the vibration signals collected from the engine. The developed system was capable of detecting any combination of the selected faults when using a set of new vibration signals.

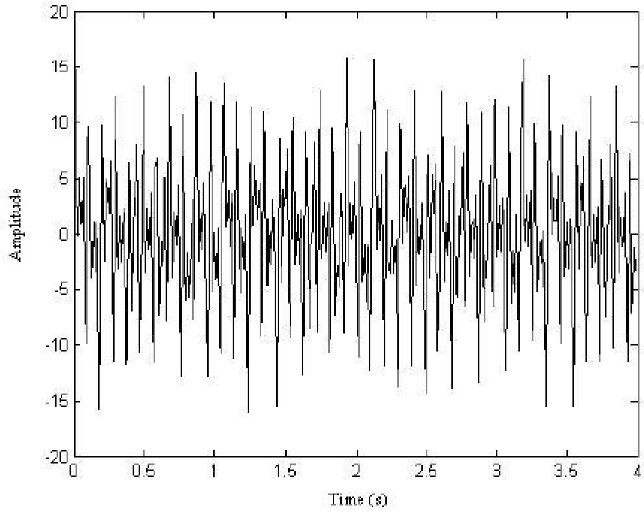

Figure 1: Vibration pattern for the faults combination of bearing, parallel misalignment and angular misalignment

Revised Version Manuscript Received on November 02, 2015.

N. D. Liyanagedera Department of Computing & Information Systems, Wayamba University of Sri Lanka Kuliyapitiya, Sri Lanka.

A. Ratnaweera Department of Mechanical Engineering, University of Peradeniya, Peradeniya, Sri Lanka.

D. I. B. Randeniya, Department of Mechanical Engineering, University of Peradeniya, Peradeniya, Sri Lanka.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

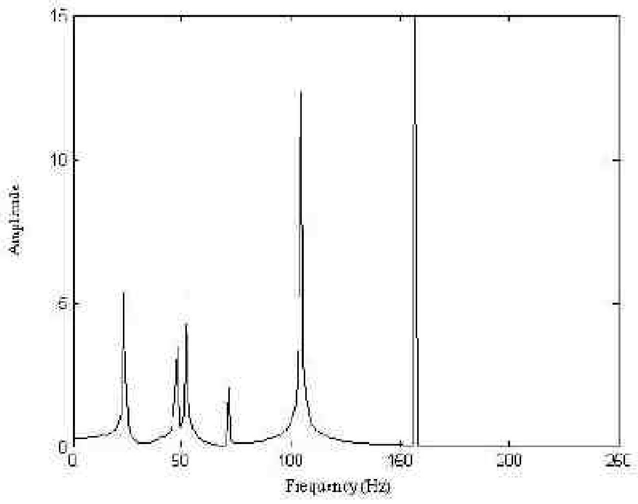

Figure 2: Fourier transformed vibration pattern for the faults combination of bearing, parallel misalignment, angular misalignment

There are many types of neural networks with different types of algorithms that can be used for a selected dataset. [9] But the performance of each algorithm will vary and depends on the characteristics of the dataset and how each algorithm seeks the optimal solution. This produces the need for finding the most suitable neural network with the highest performance for a given data set. [10],[11] As the next step in the research, engine vibration signals were tested for various types of neural networks to compare the performances and to find the most suitable neural network to detect faults in a combustion engine. Among many neural networks tested, four networks outperformed the others and those four were taken for the final testing. Resilient Backpropagation, Scaled Conjugate Gradient, Levenberg-Marquardt and BFGS Quasi-Newton were the four neural networks used for the final performance comparison. [12] All four networks were tested for fault detection of the combustion engine. The performance of each of the neural network was measured using Mean Squared Error (MSE), Time, Sum Squared Error (SSE), Mean Absolute Error (MAE) and Epoch. According to the results obtained, Resilient Backpropagation had the best performance.

Resilient backpropagation neural network [13] is an improved version of a normal back propagation algorithm and currently there are several versions of resilient backpropagations. Although there are slight differences in the implementations, major problem solving mechanisms are similar in all those algorithm types. Although resilient backpropagation works very efficiently under certain conditions, practical use is limited due to the complexity of the implemented algorithm. Compared to other neural networks, resilient backpropagation is fairly recently introduced and some pre assumed constant values are used in the implementation which are introduced by the people who published this neural network. Namely decrease factor $[\eta^{-}]$ and increase factor $[\eta^{+}]$ are two such constants used in the algorithm and in this part of the research we concentrate on those two constants. We have a question about these constant values because, although these values can be directly used, if the algorithm user wants, he can define his own set of constant values which will be used to update the weights. [14] This research checks for the most efficient set of constant values to be used in the algorithm to update the weights, by studying the performance of the algorithm for instances where differently weighted constant values were used. This is possible to be done based on previously gained knowledge. That is, the possibility of using a neural network to handle engine vibration signals and resilient backpropagation network performing better than many other tested neural networks for the selected data.

II. RESILIENT BACKPROPAGATION NEURAL NETWORK

Resilient backpropagation neural network is practically more useful when handling a network structure with a large number of input nodes and relatively a small number of output nodes. [12] Because of the large number of input nodes, a large number of weights will be there to be updated (Equ 1) and other neural networks have practical limitation due to the large internal matrices they need to create. When only a few number of outputs are available, heavy calculations done in other algorithms are insignificant to the final result.

Resilient backpropagation algorithm will avoid these heavy useless calculations by only considering the sign of the derivative and it will not consider the magnitude like some other algorithms. (Equ 4) Based on this sign,

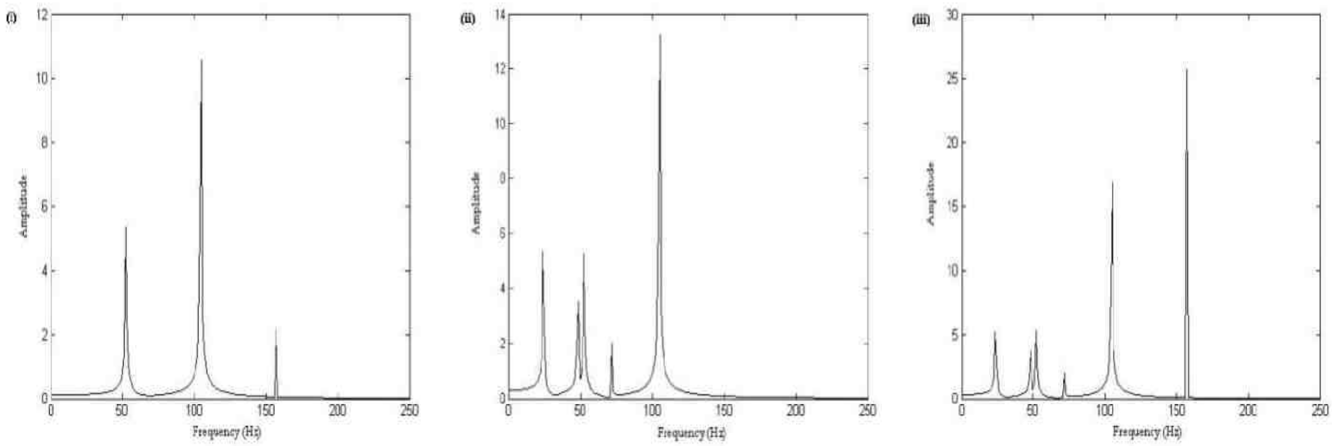

Figure 3: Comparison of Fourier transformed vibration patterns for the faults combinations of, (i) parallel misalignment (ii) bearing & fan imbalance, (iii) bearing, fan imbalance & angular misalignment

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

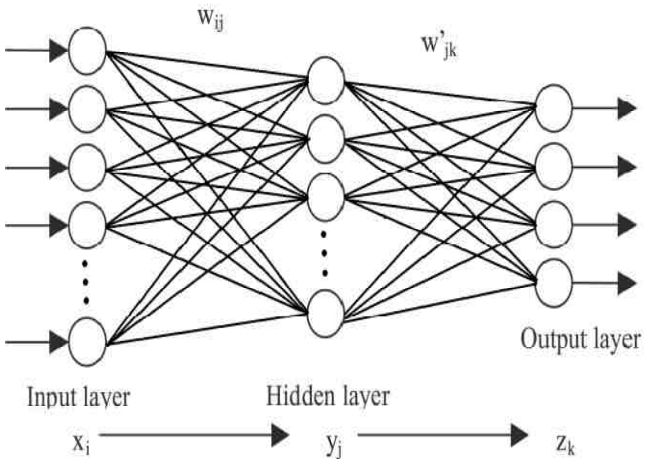

Figure 4: Artificial Neural Network model.

the weight update will be done and for the two instances, two constant values are considered when the calculations are done to determine the amount that each weight needs to be updated. (Equ 2) These two constants are known as increase factor $[\eta^+]$ and decrease factor $[\eta^-]$. If the sign of the gradient does not change in the next step, increase factor will be used to update the weights and if there is a change in the sign of the gradient in the next step, decrease factor will be used to update the weights. (Equ 6)

Although increase factor and decrease factor are two constants and their values are defined at the beginning, the user has the freedom to initialize these two values according to the needs of the system to get an optimal solution. Because, all the weight updates are done based on these constants, it's important to select the best values for these two constants to get an optimal solution form the neural network. This research test on this condition and try to check how the neural network works for different types of constant values.

Resilient back propagation learning rules are given bellow and it shows the procedure used in each step to update the weights in the neural network. [12], [13]

$$wij(t+1)=wij(t)+\Delta wij(t) \quad (1)$$

In regular gradient descent,

$$\Delta wij(t) = -\eta \frac{\partial E}{\partial w_{ij}}(t) \quad (2)$$

With momentum;

$$\Delta wij(t) = -\eta \frac{\partial E}{\partial w_{ij}}(t) + \mu \Delta wij(t-1) \quad (3)$$

Weight step rule for the Resilient Backpropagation algorithm (considers only the sign of the gradient and its update value)

$$\Delta wij(t) = \begin{cases} +\Delta ij(t),\ if\ \frac{\partial E}{\partial w_{ij}}(t) > 0 \\ -\Delta ij(t),\ if\ \frac{\partial E}{\partial w_{ij}}(t) < 0 \\ 0, otherwise \end{cases} \quad (4)$$

Exception;

$$\Delta wij(t) = -\Delta wij(t-1)\ ,\ if\ \frac{\partial E}{\partial w_{ij}}(t-1).\frac{\partial E}{\partial w_{ij}}(t) < 0 \quad (5)$$

Learning rule for the Resilient Backpropagation algorithm

$$\Delta ij(t) = \begin{cases} \eta^+.\ \Delta ij(t-1)\ ,\ if\ s_{ij} > 0 \\ \eta^-.\ \Delta ij(t-1)\ ,\ if\ s_{ij} < 0 \\ \Delta ij(t-1)\ , otherwise \end{cases} \quad (6)$$

Where

$$s_{ij} = \frac{\partial E}{\partial w_{ij}}(t-1) . \frac{\partial E}{\partial w_{ij}}(t)$$

η^- = 0.5 & η^+ = 1.2; are used for the general case but, user can define these values accordingly for special cases.

As the above learning rule states, if the sign of the gradient does not change in the next step, the update is done by 1.2 and if there is a change in the sign, the update is done by 0.5 values. In each step, the weights are updated base on this rule as defined by the algorithm developer. [13] But we have tested the performance of resilient backpropagation neural network for different values of increase factor and for different values of decrease factor. This would show the most suitable values to be used for increase factor and decrease factor.

III. MATERIALS AND METHODS

The vibration signal data collected from the combustion engine was used in these experiments. The data was collected for four types of engine faults namely bearing, fan imbalance, parallel misalignment and angular misalignment and for all possible combinations of faults, the vibration signals were obtained. Altogether there were 16 fault combinations with their corresponding vibration signals and a relationship between these two datasets was created. [8] Because of the nonlinear relationship between these two datasets, neural network models were used due to the practical impossibility in using statistical techniques to properly create boundaries to obtain 16 different groups corresponding to the 16 different fault types. When testing for performance comparison, resilient backpropagation outperformed several other tested neural networks. [12] All these tests showed a quality output provided by resilient backpropagation when engine vibration signals were used to detect fault in the engine. The close performance match between the data and the resilient backpropagation neural network allowed us to go for the next part of the research. Earlier, neural network was used to test the data. But now, the data will be used to test some characteristic conditions of the neural network.

As shown under the neural network description above, resilient backpropagation algorithm consists of two constant values namely decrease factor $[\eta^-]$ and increase factor $[\eta^+]$. (Where $0 < \eta^- < 1 < \eta^+$) For the algorithm to work properly, these two constant values need to be in the given range. The algorithm developers states η^- = 0.5 & η^+ = 1.2 can be used to get an optimal result [13] but, any results are

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

not shown to indicate if these values are the best values. For the engine vibration signal data, these two constant values were tested in the given range and the performances were checked. The test was conducted for two instances where in one instance, the decrease factor $[\eta^-]$ was kept as a constant value and the increase factor $[\eta^+]$ was changed in the range starting from 1.05 to 1.7. In the next instance, the increase factor $[\eta^+]$ was kept as a constant value and the decrease factor $[\eta^-]$ was changed in the range from 0.1 to 0.9. The algorithm performance was measured using mean square error (MSE), epoch & time. Lower values of these performance indicators will indicate that the algorithm performed well in those instances and increase factor $[\eta^+]$ and decrease factor $[\eta^-]$ values used in those instances can be considered to be good values that can be used in resilient backpropagation algorithm.

Testing of all instances of the algorithm was done using the engine vibration signals, where fault identification was the main target. The whole procedure of converting the vibration signal data into a usable format, developing the multilayer neural network and testing the algorithm, consists of lengthy descriptions and they are shown in papers. [8][12] In past work, constant values were used for both increase factor $[\eta^+]$ and decrease factor $[\eta^-]$ and fault detecting ability was checked. But in this part of the research, either increase factor $[\eta^+]$ or decrease factor $[\eta^-]$ was changed in a selected range and the corresponding performances were measured. There are two advantages in doing this test. The first advantage is, the result can be used to further improve the development of the engine fault detecting system. The second advantage is, the results will indicate what values are most suitable to be used for increase factor $[\eta^+]$ and decrease factor $[\eta^-]$. This result will be useful for someone who plans to use resilient backpropagation algorithm and having a problem about the most optimal values to be used for increase factor $[\eta^+]$ and decrease factor $[\eta^-]$.

Table 1. Performance Indicator Results of Resilient backpropagation neural network for changing Decrease factor with constant Increase factor. (1.2)

Decrease factor	MSE	Epoch	Time (s)
0.1	5.99×10^{-12}	186	22
0.2	3.11×10^{-12}	156	18
0.3	4.30×10^{-12}	104	13
0.4	3.30×10^{-12}	81	10
0.5	7.89×10^{-12}	63	7
0.6	1.28×10^{-11}	53	6
0.7	1.24×10^{-9}	45	5
0.8	5.88×10^{-2}	34	4
0.9	8.96×10^{-2}	29	3

Table 2. Performance Indicator Results of Resilient backpropagation neural network for changing Increase factor with constant Decrease factor. (0.5)

Increase factor	MSE	Epoch	Time (s)
1.05	9.41×10^{-12}	109	12
1.1	5.60×10^{-12}	96	11
1.2	8.02×10^{-12}	75	9
1.3	7.60×10^{-12}	60	7
1.4	1.64×10^{-9}	40	5
1.5	2.90×10^{-2}	35	4
1.6	7.38×10^{-2}	28	3
1.7	8.71×10^{-2}	19	2

IV. RESULTS AND DISCUSSION

Engine vibration signals and the corresponding fault combinations were used to train, test and validate the resilient backpropagation neural networks. Performances of the neural networks were measured using epoch, Mean Square Error (MSE) and time. The testing consists of two parts where, the performance of the neural network relative to the increase factor and decrease factor were measured separately.

In the first part, the behavior of the neural network was tested for changing decrease factor $[\eta^-]$. To do this measurement properly, the increase factor $[\eta^+]$ was kept in a constant value of 1.2. For the performance measure, mean square error (MSE), epoch and time were taken for different values of decrease factor and shown in the table. [Table 1] Plots of each of these values against the decrease factor were also obtained. Graph [Figure 5] shows the plot between epoch and decrease factor. Mean square error (MSE) changes happen in a large range and to properly plot it, log(MSE) was considered and Graph [Figure 6] shows the plot between log(MSE) and decrease factor. Graph [Figure 7] shows the plot between time and decrease factor. Lower values in these three plots will indicate the best performance of the neural network and the corresponding decrease factor for this instance can be observed. To

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

properly observe the behavior of these performance measurements, the three graphs were plotted in a single graph. [Figure 8] Because three different types of data were plotted in a single graph, without considering each data type, all three of them were rescaled onto a range where the minimum value = 10 and maximum value = 90 were used and plotted in the range from 0 to 100.

In the second part, the behavior of the neural network was tested for changing increase factor [η^+]. To do this measurement properly, the decrease factor [η^-] was kept in a constant value of 0.5. For the performance measure, mean square error (MSE), epoch and time were taken for different values of increase factor and shown in the table. [Table 2] Plots of each of these values against the increase factor were also obtained. Graph [Figure 9] shows the plot between epoch and increase factor. Mean square error (MSE) changes happen in a large range and to properly plot it, log(MSE) was considered and Graph [Figure 10] shows the plot between log(MSE) and increase factor. Graph [Figure 11] shows the plot between time and increase factor. Lower values in these three plots will indicate the best performance of the neural network and the corresponding increase factor for this instance can be observed. To properly observe the behavior of these performance measurements, the three graphs were plotted in a single graph. [Figure 12] Because three different types of data were plotted in a single graph, without considering each data type, all three of them were rescaled onto a range where the minimum value = 10 and maximum value = 90 were used and plotted in the range from 0 to 100.

The graph [Figure 8] indicates how resilient backpropagation neural network behaves for different values of decrease factor. When considering the minimum values of the three plots, the priority should be given to considering the minimum value of mean square error (MSE). This is because, the plot is done using log(MSE) and for a small change in a log value the actual value change is very high. Because of this condition, best value for the decrease factor should be selected from the range where log(MSE) is at the lowest level.

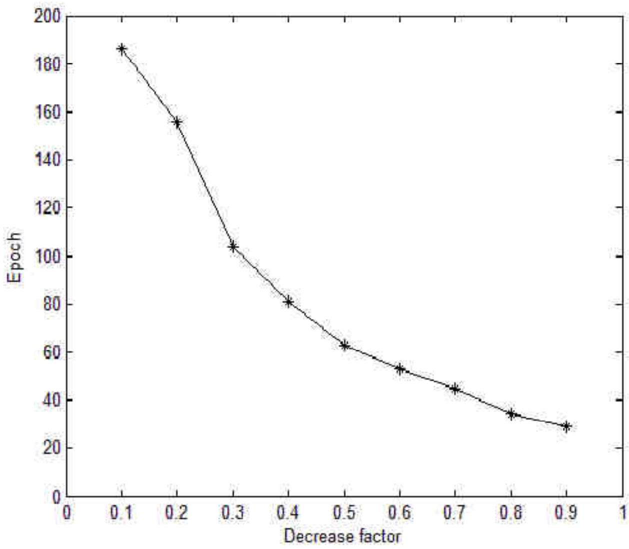

Figure 5: Epoch vs Decrease factor for constant Increase factor.

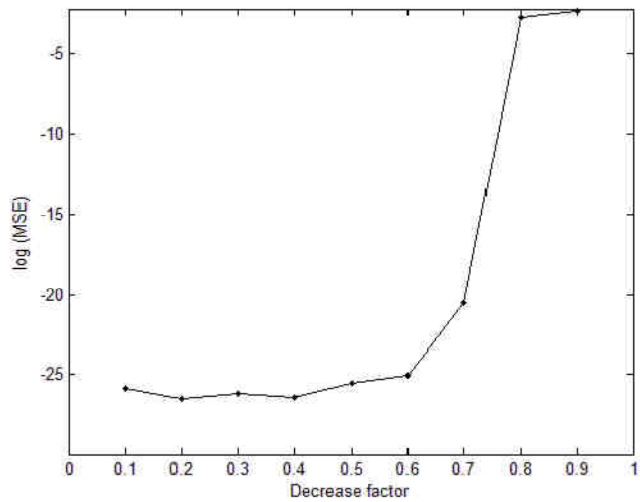

Figure 6: log(MSE) vs Decrease factor for constant Increase factor.

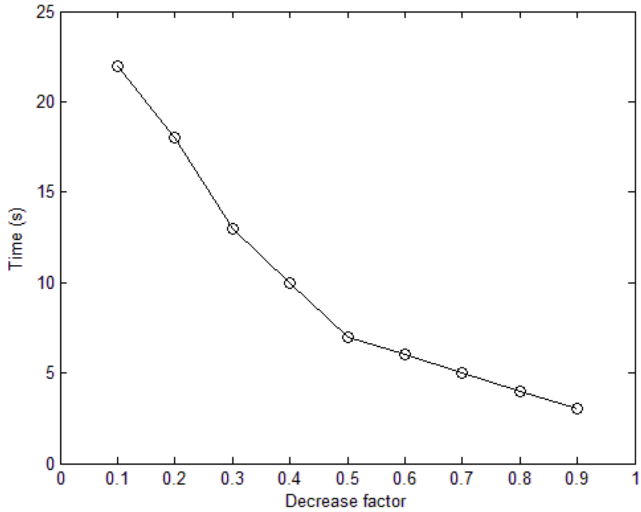

Figure 7: Time vs Decrease factor for constant Increase factor.

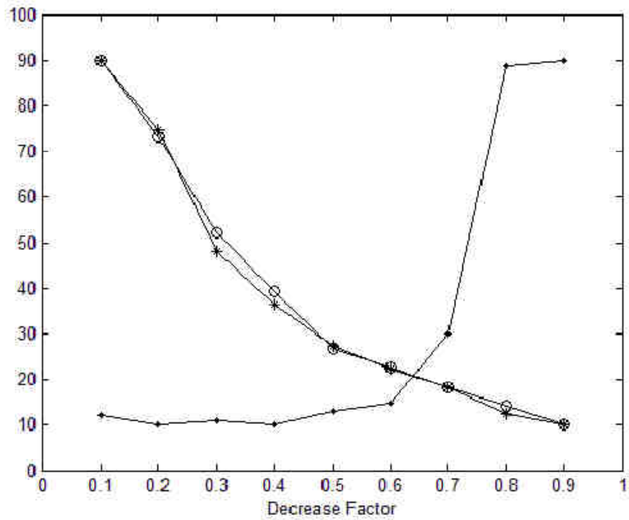

Figure 8: Epoch, log(MSE), Time vs Decrease factor for constant Increase factor.

Based on the log(MSE) vales, the decrease factor value in the range 0.1 to 0.6 looks suitable. Now we can increase the accuracy of selecting the best decrease factor value by looking at the plots of epoch and time. The plot of epoch

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

and the plot of time, both have a similar behavior continually decreasing in value from start to end. But based on the results of log (MSE) we only check the range 0.1 to 0.6. In this range, the minimum values for epoch and time can be obtained when considering the decrease factor in the range from 0.5 to 0.6. When considering all three performance indicators, we can see that the resilient backpropagation algorithm has the best performance when the decrease factor is in the range of 0.5 to 0.6.

The graph [Figure 12] indicates how resilient backpropagation neural network behaves for different values of increase factor. When considering the minimum values of the three plots, the priority should be given to considering the minimum value of mean square error (MSE). This is because, the plot is done using log (MSE) and for a small change in a log value, the actual value change is very high. Because of this condition, best value for the increase factor should be selected from the range where log(MSE) is at the lowest level. Based on the log(MSE) vales, the increase factor value in the range 1.05 to 1.3 looks suitable. Now we can increase the accuracy of selecting the best increase factor value by looking at the plots of epoch and time. The plot of epoch and the plot of time, both have a similar behavior continually decreasing in value from start to end. But based on the results of log(MSE) we only check the range 1.05 to 1.3. In this range, the minimum values for epoch and time can be obtained when considering the increase factor in the range from 1.2 to 1.3. When considering all three performance indicators, we can see that the resilient backpropagation algorithm has the best performance when the increase factor is in the range of 1.2 to 1.3.

V. CONCLUSION

The objective of this study is to test the performance of resilient backpropagation neural network for different values of increase factor and decrease factor. This will allow us to identify the most suitable values to be used for the increase factor and decrease factor, allowing us to get the maximum performance out of resilient backpropagation algorithm. A combustion engine fault detection system was used as the test system for this. Engine vibration signals and four types of engine faults were used as the data for this system. As the performance indicators, mean square error, time and epoch were used. When considering all three performance indicators, we can see that the resilient backpropagation algorithm had the best performance when the decrease factor $[\eta^{-}]$ was in the range of 0.5 to 0.6 when increase factor was kept as a constant. Similarly, when decrease factor was kept as a constant, the resilient backpropagation algorithm had the best performance when the increase factor $[\eta^{+}]$ was in the range of 1.2 to 1.3. Increase factor and decrease factor are values used to update the weights in each step in the neural network. The weight update is the main activity in a neural network to obtain the optimal solution. This indicates the value of using the best increase factor value and decrease value when implementing resilient backpropagation.

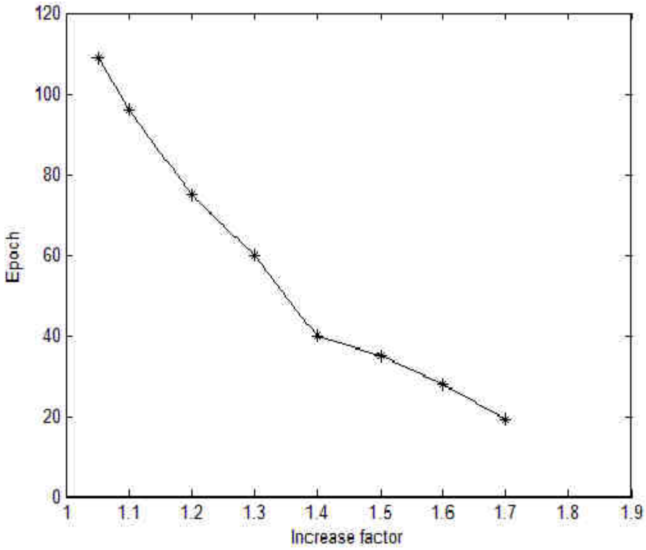

Figure 9: Epoch vs Increase factor for constant Decrease factor.

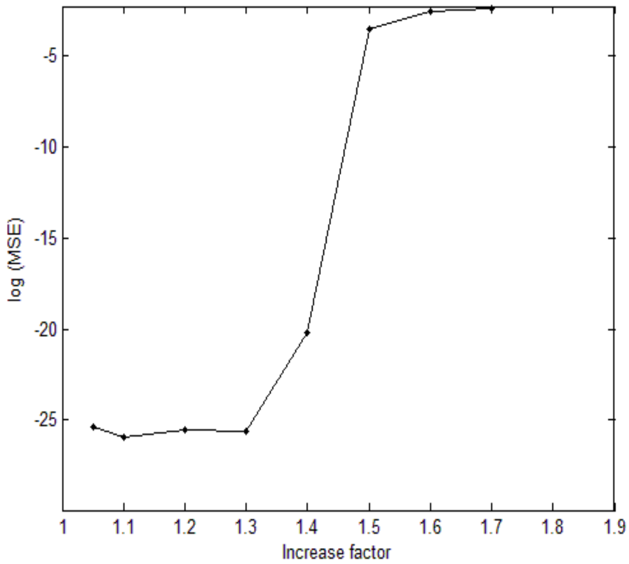

Figure 10: log(MSE) vs Increase factor for constant Decrease factor.

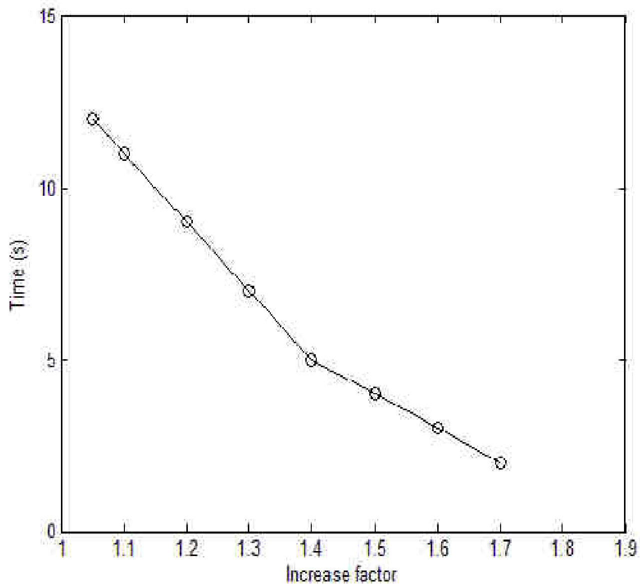

Figure 11: Time vs Increase factor for constant Decrease factor.

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication Pvt. Ltd.

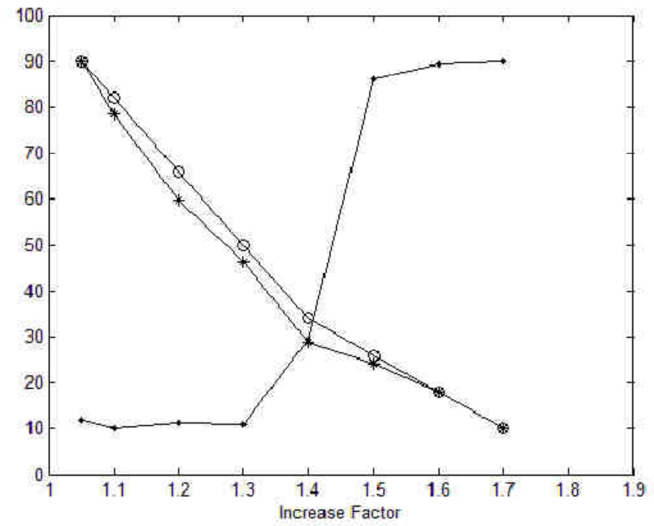

Figure 12: Epoch, log(MSE), Time vs Increase factor for constant Decrease factor.

The results obtained in this experiment will depend on the characteristics of the engine vibration signals and on the characteristics of the algorithm used. The results obtained from this work can be very helpful for the engine fault detection system as well. When the resilient backpropagation algorithm gives its best performance, it would increase the quality and the efficiency of the engine fault detection system. And also, the knowledge obtained about the values to be used as increase factor and decrease factor can be used by anyone who plans to use resilient backpropagation neural network in their work, if they have a doubt about what values to be used. For such people this result can be used as a starting guideline and go for necessary modifications based on the characteristics of their own data.

REFERENCES

1. T. Denton, Advanced Automotive Fault Diagnosis, 2nd Edition, Elsevier Ltd, 2006.
2. S. Klinchaeam, P. Nivesrangsan, M. Lokitsangthong, Condition Monitoring of a Small Fourstroke Petrol Engine using Vibration Signals. KMITL Sci. Tech. J. Vol. 9 No. 1 Jan. - Jun. 2009.
3. S. Ramezani , A. Memariani , A Fuzzy rule base system forfault diagnosis, using oil analysis results, International journal of industrial engineering and production research, June 2011, volume 22, number 2, pp 91-98.
4. K. Wang , Neural network approach to vibration feature selection and multiple fault detection for mechanical systems, International Conference on Innovative Computing, Information and Control, 2006 IEEE.
5. K. Suzuki, Artificial Neural Networks - Industrial and Control Engineering Applications,InTech, 2011, ISBN 978-953-307-220-3.
6. H. Su, K. Chong, R. R. Kumar Vibration signal analysis for electrical fault detection of induction machine using neural networks, Springer-Verlag London Limited 2011.
7. R. Ahmed, M.E.l. Sayed, S.A Gadsden, J. Tjong, S.Habibi, Automotive Internal-Combustion-Engine Fault Detection and Classification Using Artificial Neural Network Techniques. IEEE Transactions on vehicular Technology, Vol. 64, No. 1, January 2015.
8. N. D. Liyanagedera, A. Ratnaweera, D. I. B. Randeniya, Vibration Signal Analysis for Fault Detection of Combustion Engine Using Neural Network, Industrial and Information Systems (ICIIS), 2013 8th IEEE International Conference.
9. X. Pan, B. Lee, C. Zhang, A Comparison of Neural Network Back propagation Algorithms for Electricity Load Forecasting. Intelligent Energy Systems (IWIES), 2013 IEEE International Workshop.
10. I. Vilovic, N. Burum, A Comparison of MLP and RBF Neural Network Architectures for Location Determination in Indoor Environments. Antennas and Propagation (EuCAP), 2013 7th European Conference, IEEE.
11. C. Juan, G. Higuera, G. Alba, A. Bernal, M. Julian, F. Angel, A Comparison of Neural Networks to Detect Failures in Micro-electro-mechanical Systems , Electronics, Robotics and Automotive Mechanics Conference, 2010, ISBN: 978-0-7695-4204-1,pp: 191-196.
12. N. D. Liyanagedera, A. Ratnaweera, D. I. B. Randeniya, Performance Comparison of Four Neural Networks to Detect Faults in a Combustion Engine, NNGT Int. J. on Artificial Intelligence, Vol. 2, Feb 2015.
13. M. Riedmiller , H. Braun, A direct adaptive method for faster backpropagation learning: the RPROP algorithm, IEEE International Conference on Neural Networks, 1993.
14. X. Wang, H. Wang, G. Dai, Z. Tang, A Reliable Resilient Backpropagation Method with Gradient Ascent, Computational Intelligence, International Conference on Intelligent Computing, ICIC 2006, Kunming, China, August 16-19, 2006.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.