International Journal of Manufacturing and Material Processing
Vol. 1: Issue 1
www.journalspub.com

A Comprehensive Study on Optimization of Neural Networks and Global Minimization

M. Aravindh, C. Velmurugan*
Department of Mechanical Engineering, Kumaraguru College of Technology, Coimbatore, India

Abstract
The neural network toolbox is one of the commonly used, commercially available and powerful software tool for constructing new networks and it forms a inter linkage between functionalities and their respective objective function. The software is user-friendly, permits flexibility and convenience in interfacing with other toolboxes in the same environment to develop a full application and has a wide range of scope in making models with linear and non-linear functions, without any assumptions and applied in almost every field of science and engineering. In addition, it underlies in one directional linear function with and without hard limit, and possesses the responsibility of radial and triangular biasing and suitable for competitive soft max functions. A wide variety of training and learning algorithms are supported through neural networks. The functions used in the neural networks are as same as model of brain and nervous system which are highly parallel and possess high speed information resemble the brain and a super computer. The functionalities in neural networks have very simple principles and possess very complex behaviors. In this paper, an attempt is made to enhance effortless operation of neural networks in a systematic procedure through MATLAB for optimizing and global minimizing.

Keywords: global minimization, MATLAB, neural networks, optimization, training

***Corresponding Author**
E-mail: m.aravindhmech27@gmail.com

INTRODUCTION

An optimization function is defined as the system which possesses a function for getting objective values with a valid constraint equation set and variables that defines a function. The objective of any optimization is that find the variables that lead to an optimal value to find (either minimum or maximum), while satisfying all the constraints. [1–3]

Learning of problem domains for neural network is considered by the factor when a Input is highly-dimensional discrete or real-valued (e.g. raw sensor input) and output in the form of discrete or real valued which has a vector of values and considered as a target function, Humans need not to interpret during the results.[4]

Neural networks can be trained using local, gradient-based procedures. Such methods frequently find suboptimal solutions as local minima. Optimization of neural network structures and global minimization methods is applicable for cost functions have strong influence on all aspects of network performance. [5]

The structure of neurons is considered with the input layers, output layers and hidden layer. Neuron acts as an input signal when it exceeds a certain amount in a short time period. Synapses or connection may vary in strength that depends upon the interconnection between function and objective function. Good connections allowing a signal which possess high strength. Slight connections may allow a weak signal which may not be effective as discussed in Figures 4 and 5 [6]

It has an extensive range of purpose such as processing data for exact analysis, optimizing of systems, maximization of profits, minimization of losses and errors, interpolation of models, clustering of different systems, Forecasting and prediction of futuristic values, proficient estimation. [7]

Training, control identification, uniqueness revealing and sequencing, decision making, approximation of functions and differentiate the performance of existing system with the new model can be done with neural networks.[4, 5]

SYSTEMATIC PROCEDURE FOR NEURAL NETWORK

NN implementation is a process that results in bringing best design percentages of classification accuracy and mean square error are used for accuracy as shown in Figure 1. [1]

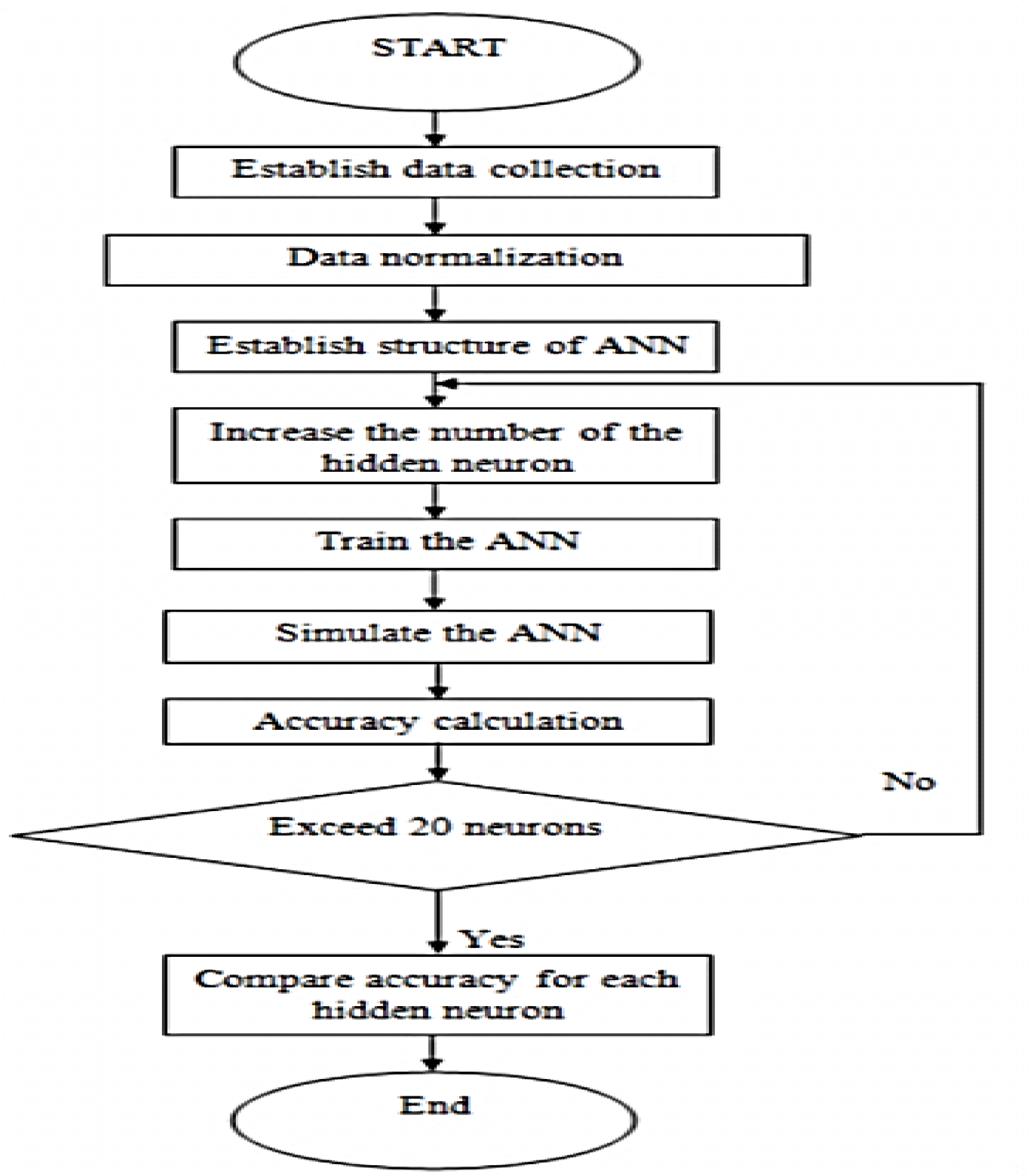

Fig. 1. *Systematic Representation of Neural Network.*

STEPS OF NN IMPLEMENTATION

Neural networks have an organized step by step procedure for implementation as shown in Figure 2. [1]

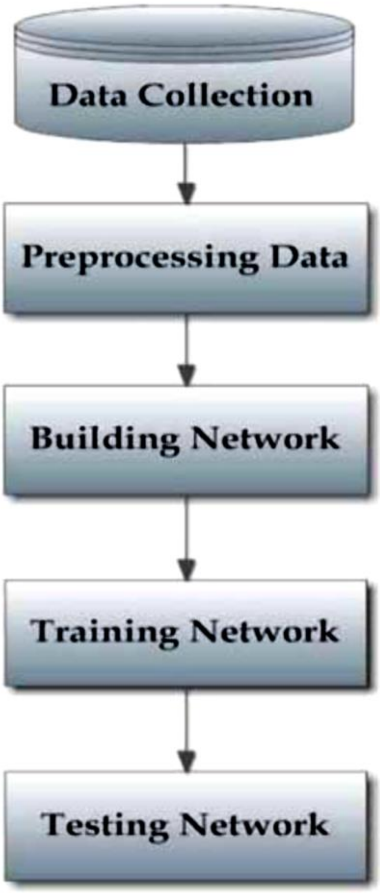

Fig. 2. *Implementation Procedure of Neural Network.*

The inputs are represented as the variables of x and weights are added to inputs to ensure best value feed and outputs were obtained in neuron output and hidden layers ensure the system with rich functionalities to find optimal solution (Figures 3, 4).

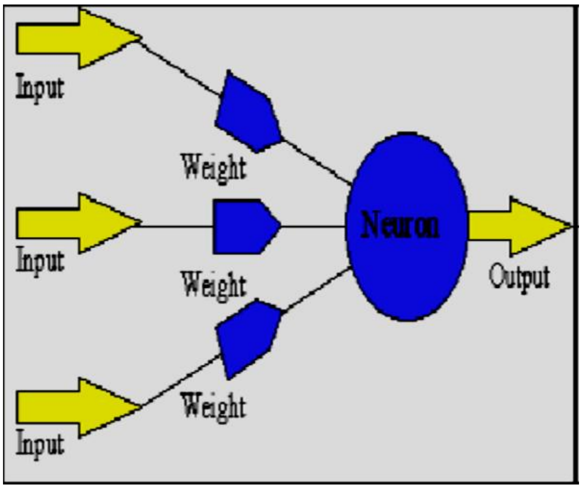

Fig. 3. *The Representation of Neural Network.*

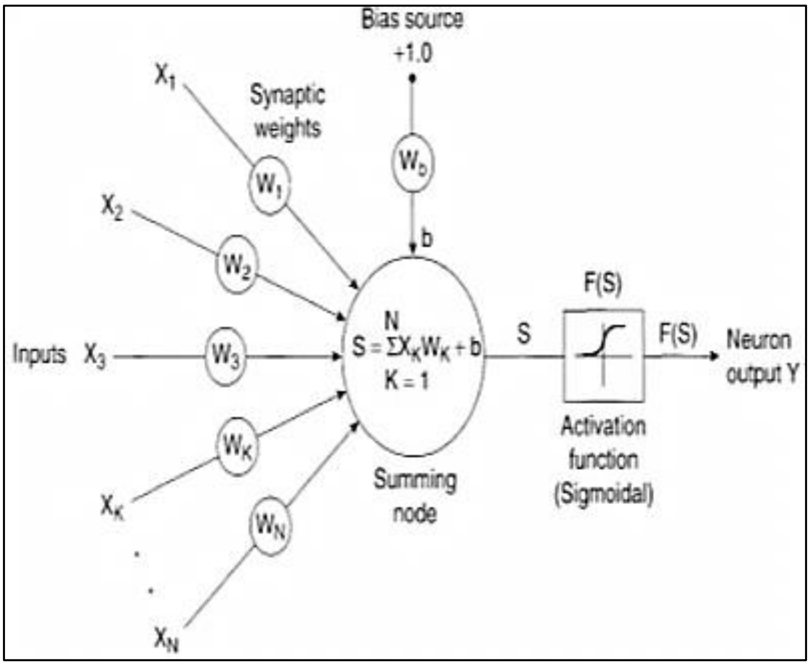

Fig. 4. The Topology of Neural Network.

COMPONENTS OF NEURAL NETWORKS

NN incorporate the two basics fundamental component for neurons (nodes) and weight (synapses) which forms the basics structure of NN (Figure 5). [1]

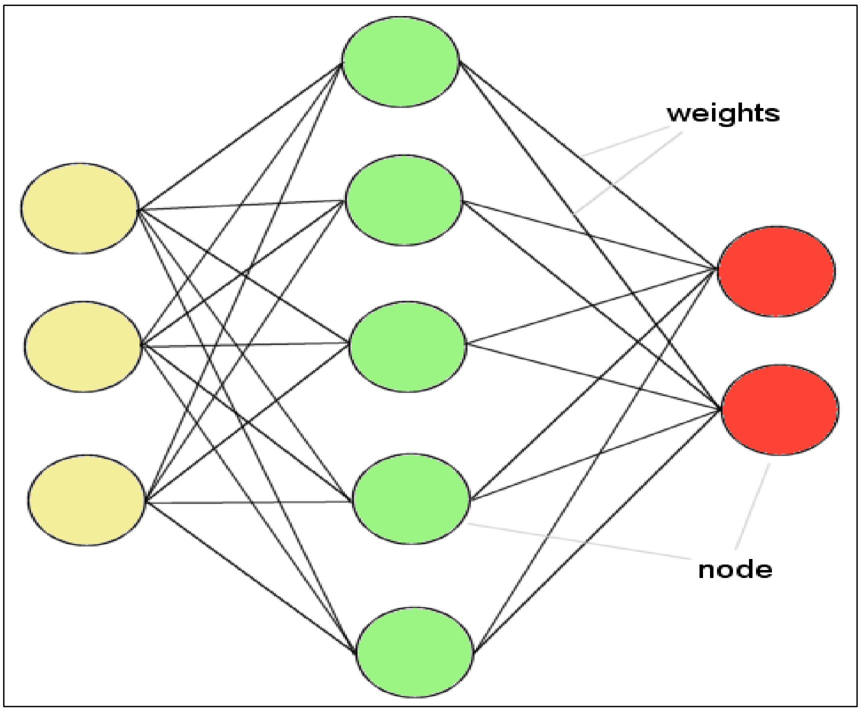

Fig. 5. The Neural Network Components.

OPTIMIZATION USING NEURAL NETWORKS

Software Employed MATLAB 2014 A

Optimization can be done through various systematic procedures such as genetic algorithm, ant colony algorithm and particle swarm algorithm. But the easiest and powerful way to optimization can be done only through neural network. It is one among the non-traditional approaches for optimization. Any linear functions and their regression equations can be formed, trained, adapted and optimized for achieving best fit.

STEP BY STEP OPTIMIZATION PROCEDURE

1. Open neural network tool command window dialog box by feeding NN tool graphical user interface.
2. Feed input and output in the workspace and import them in their respective arena. Where input is the combination of factor and its level (–2, –1, 0, 1, 2) Output is the experimental value of heat treatment composites.
3. Create a network with the following parameters. Building a network is not an easy task. The type of network employed in the system is feed forward back prop and training function for systematic task can be done through trainlm, adaptive techniques in the learning options are marked as learngdm. Measurement of functions and their coefficient levels are selected as performance function: mean square error (MSE) and number of layers is selected as 2, number of neurons is set up to 1000 for best clustering values. Transfer of functions from an input value to their respective output layer is marked as transfer function: purelin. Thus, network is created with specified inputs and outputs as shown in Figure 6.
4. Train function – train the function for the respective inputs and outputs with the following parameters: epochs – 1000, data division – Random (dividerand), time – Infinite, training – Levenberg Marquardt, goal – 0, minimum grade – $1e^{-07}$, max fail – 1000, performance – MSE. The accuracy of the result depends upon proper training.
5. Plot the graph for simulating the network. The accuracy of the system can be evaluated with the graph of regression, performance and training function as shown in Figures 3–5. The linearity function with proportional values plotted in the graph gives us the best optimal result. The change in linearity the accuracy of the result gets diversified. Adaption of network is done when the linearity gets diversified beyond the certain limit and reinitializing weights for input parameters are set. The optimal values of the heat treatment parameters are displayed at data manager dialog box and corresponding errors values are also noted for optimality diversification.
6. Open the network fitting tool and save the result in the form of transfer, weight, process and control functions. The accuracy of result depends on the training function, performance plotting and validation as shown in Figures 7–11.

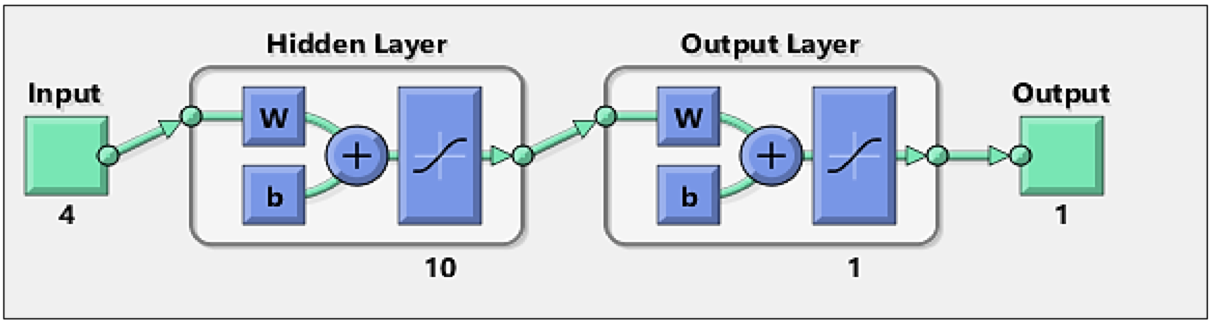

Fig. 6. *Creation of a Network.*

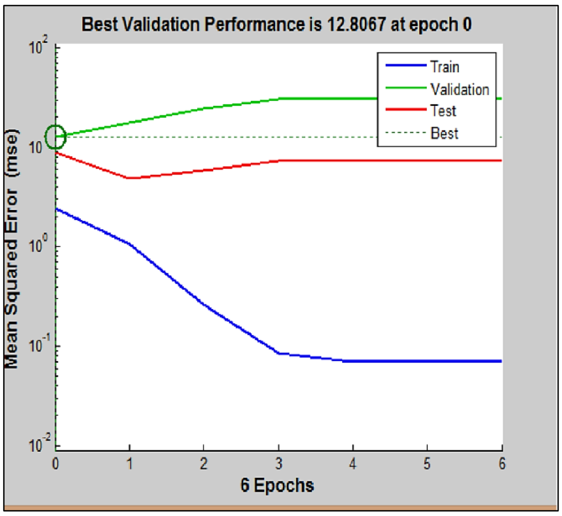

Fig. 7. *Training Plot With Linearity and Validation.*

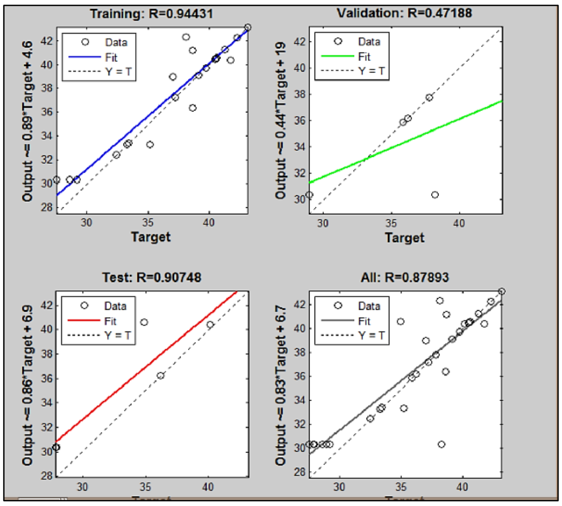

Fig. 8. *Training Plot With Best Fit.*

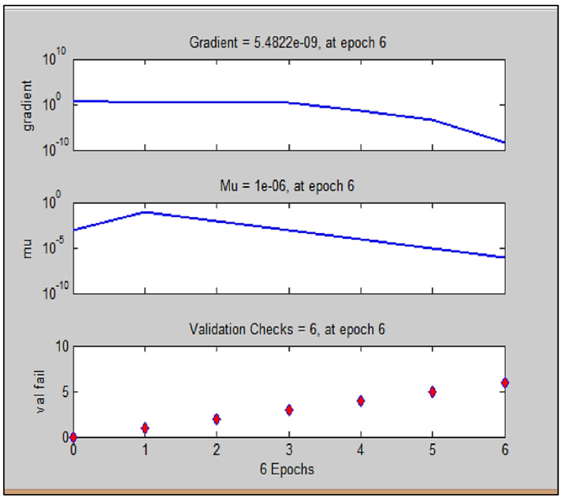

Fig. 9. *Performance Plot With Best Validation.*

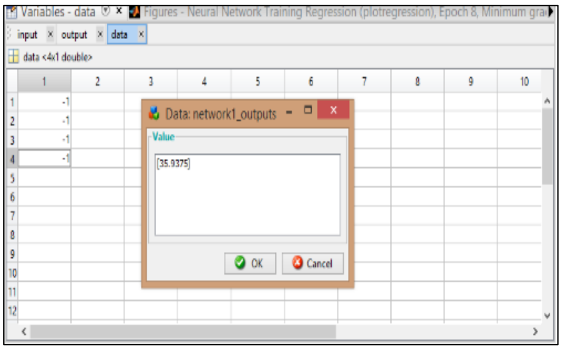

Fig. 10. *Data Network Sample Output With Optimized Value for a Factor (–1, –1, –1, –1).*

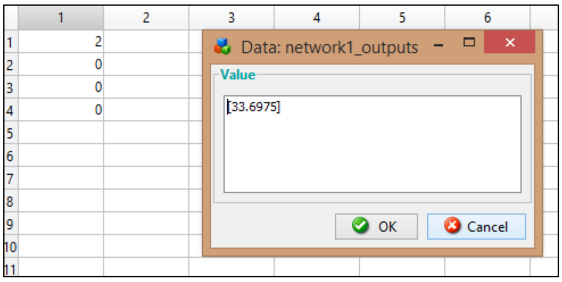

Fig. 11. *Data Network Sample Output With Optimized Value for a Factor (2, 0, 0, 0).*

Parameters Considered for Training.[6]

Backtracking search – best performance is obtained only when satisfactory decline of backtrack that helps search routine linearity that starts with a multiplier of 1 and search functions.

Epoch – a network which has the set of preplanned training (input and/or target) vectors to and the calculation to possess best presentation.

Levenberg–Marquardt – The system which trains a function of neural network in a rapid manner than the usual back propagation gradient method and highly validated.

MSE function – The calculation of performance between the network outputs and the target outputs in the form of square error.

Training – The set of systematic procedure whereby a network is set to standards to perform a particular job.

Global minimum – The lowest value in a function over the entire range of its input parameters. The global minimum can be calculated by using gradient method and outputs are in optimized form.

Local minimum – The input values with limited range possess minimum of a function. A local minimum may not be the global minimum (Figure 12).

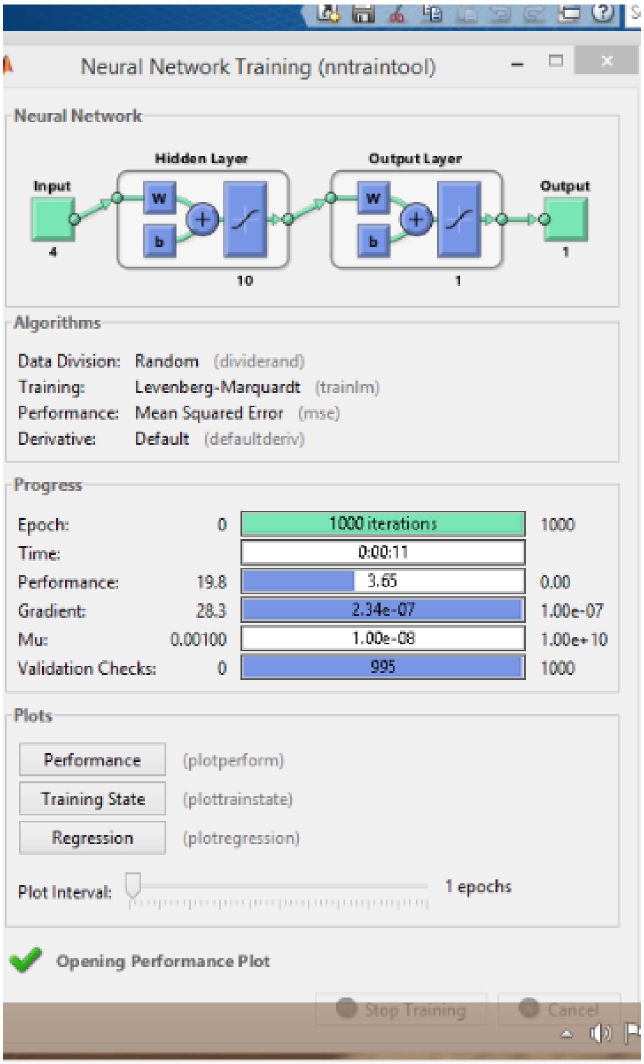

Fig. 12. *Data Network for Training.*

Calculation on the Reliability of Optimized Values [6]

The reliability of the optimized values is calculated to check the deviations in the values using conformity tests using a formula,

Error = actual value–optimized value/optimized value/100

Where, the actual value is considered as experimental value, optimized value is considered as neural network interpolated values, the values of error of experimental and optimized values should not exceed more than 1

CONCLUSION

The step by step procedure for optimizing has been illustrated in this paper and an attempt is done to help researcher for quick access for optimizing the experimental values and to obtain optimal value the solution obtained in this technique is highly safe and accurate. Optimizing technique such as simulated annealing, genetic algorithm, particle swarm algorithm, and ant colony algorithm was extremely monotonous but NN provides easier way of optimizing and it is highly flexible. No special training for feeding program and the program is inbuilt in MATLAB. Researcher can utilize artificial neural networks as an estimation tool and as a method of global optimization. In modern days the most frequently used command in optimization of neural networks is back propagation. NN possess the ability to find solutions with high accuracy and function appropriation that becomes a major impact factor for popularity. Back propagation helps the investigator and researchers for best interpolation data's and it has major impact on all application oriented structures and performance with unpredictability.

(i) The advancement and analysis of neural networks to other optimization technique is significantly superior and extremely effortless and a powerful tool for obtaining results with an effective time series.
(ii) The NN provides an appropriate tool for optimization and it can be used in various applications.
(iii) Quick access to researchers to optimize experimental result.
(iv) High degree of accuracy can be obtained and quick decision making can be done.
(v) Thus the globalized search technique – NN tool can be identified as the potential solution for better result.[1,3,5,7–10]

REFERENCE

1. Demuth H., Beale M. *Neural Network Tool Box the Math Work.inc Publication – The Use with Toolbox*. 2012.
2. [WaTa96] Wang J., Takefuji Y. *Neural Computing for Optimization and Combinatory*. World Scientific Pub Co; 1996.
3. Lawduch W., Korczaz J. *Neural Networks & Global Optimization Method Suitable for Neural Networks*. 2008.
4. Lillo W.E. *Neural Network for Constrained Optimization Problem*. 1991.
5. Lagodakis M.G. *Neural Networks & Optimization Problem.* John Willey & Son's Publication; 1996.
6. Maheswaran C. Investigation on the wear behavior of aluminum-6061 and aluminum oxide hybrid composites 2014 using ANN, *Int J Curr Eng Technol*. 1(3): 363–9p.
7. Haykin S. Neural Networks-a Comprehensive Foundation, Prentice Hall; 1999.
8. Hopfield J.J., Tank D.W. "Neural" computation of decisions in optimization problem, *Biol Cybernet*. 52: 141–52p.
9. Ansari N., Hou E. *Computational Intelligence for Optimization*. Norwell, MA: Kluwer Academic Publishing Group; 1997.
10. Rao V.B., Rao H.V. *Neural Networks and Fuzzy Logic*. 2nd Edn. MIS Press; 1995, Ch. 15.