Metrics for Measurement of Node Fault in Sigmoidal FFANNs

Amit P. Singh, *Member, IACSIT*, C. S. Rai, and Pravin Chandra

***Abstract*—An important issue in the design of a neural network is the sensitivity of its output to input, and node fault. In this paper, new sensitive measures are proposed for node fault, specifically node stuck-at-zero fault. Correlation coefficient between empirical mean squared error and error due to proposed metric shows that the proposed metrics are significant metrics due to their statistical significance at 95% confidence level for node stuck-at-zero fault.**

***Index Terms*—Sigmoidal feedforward neural network, node fault, fault metric, fault tolerance.**

I. Introduction

An artificial neural network (ANNs) consists of large numbers of highly interconnected processing elements called neurons. Due to the structure of neural network, it is normally assumed to have a fault tolerant architecture [1], but it is found in literature [2]-[4] that more study is required to design fault tolerant neural network.

A fault model is proposed in [5] and [6] to study effect of faults of ANNs. Basic faults in ANNs are enumerated as:

1) Fault/Error in input.
2) Fault/Error in weights/nodes
3) Fault/Error due to noise

The fault/errors may be classified as:

1) The fault/error in input is defined as any outside disturbances, which affect the input of a network. Input error is also defined as an external fault. It means, missing a component from a input vector leads to error in output [7] or contaminated by noise.

2) Fault in internal architecture of the neural network is defined as fault in weights and nodes, fault/error in internal layer and fault/error in output nodes [2]. Further classification of these faults are characterized as stuck-at faults, which has been studied in [1],[8] and [9]. Detailed study of node fault is presented in this paper.

3) The presence of noise in weights and nodes produce error in the output of the network. Network error due to noise is studied in [10].

Node fault is studied in [11] and [12] for Radial basis function neural network and measures are proposed to Node fault is studied in [11] and [12] for Radial basis function neural network and measures are proposed to predict error due to node stuck-at-zero (NSZ) for the network. In [13], the effect of node fault in a feedforward artificial neural network (FFANN) is studied. They proposed a constraint backpropagation training method to design high degree of fault tolerance when one or two hidden nodes fail.

By incorporating constraint backpropagation training method and a technique called output node saturation, they found a network which maintains exactly the same performance as a normal network when hidden nodes fail entirely. In [14] multi--node open fault is analyzed and an approach called T3 (*Train-Test-Train*) to design a fault tolerant network is proposed.

In this paper, Section II discusses the architecture of FFANN. Section III describes sensitivity measures for node fault Section IV discusses the experiments and obtained results for node fault and the conclusion is presented in Section V.

II. Architecture of FFANN

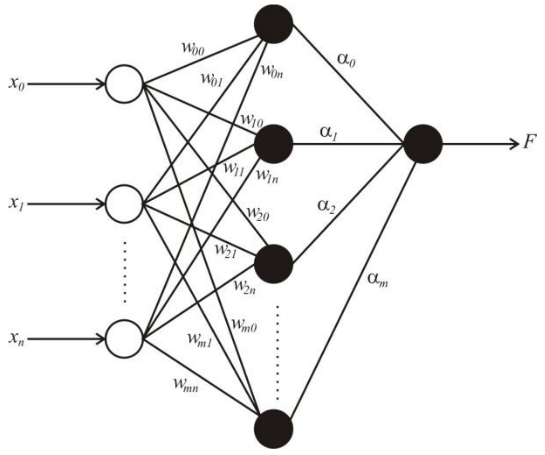

Fig. 1. Architecture of FFANN

The architecture of the feedforward artificial neural network (FFANN) is illustrated in Fig.1. A network has *n* input nodes with one hidden layer, consisting of *m* hidden layer node and a single output is selected for experiment of this work. The connection strength between *j*th input node to *i*th hidden layer node is defined as w_{ij}. So, net input to the *i*th hidden layer node is given by

$$net_i = \sum_{j=1}^{n} w_{ij} x_j + \theta_i \qquad (1)$$

where Θ_i is the threshold/bias of the *i*th hidden layer node. The output from the *i*th hidden layer node is given by

$$h_i(x) = f(net_i) \qquad (2)$$

The net input to the output node may be defined similarly to (1) as follows:

$$net = \sum_{i=1}^{m} \alpha_i h_i + \gamma = \mathbf{a.h} + \gamma \qquad (3)$$

Manuscript received October 12, 2012; revised December 20, 2012.

The authors are with the University School of Information & Communication Technology, Guru Gobind Singh Indraprastha University, Dwarka, Delhi, India (e-mail: aps.ipu@gmail.com; csrai_ipu@yahoo.com; chandra.pravin@gmail.com).

DOI: 10.7763/IJMO.2013.V3.241

where α_i represents the connection strength between the *i*th hidden layer node and the output node, while γ is the threshold/bias of the output node and a = $[\alpha_0, \alpha_1, \ldots\ldots \alpha_m]^T$, while $h = [h_0, h_1, \ldots, h_m,]^T$.

By considering threshold/bias as the weight coming from the auxiliary input node labeled as the zeroth node, (2) can be redefined as follows:

$$net = \sum_{j=0}^{n} w_{ij} x_j = \mathrm{W}_i.\mathrm{x} \tag{4}$$

where $W_i = [w_{i0}, w_{i0}, \ldots\ldots\ldots\ldots, w_{mn},]^T$ and $x = [x_0, x_1 \ldots\ldots\ldots, x_n,]^T$ and similarly introducing an auxiliary hidden node (i=0) such that h_0=1 for any input allows us to redefine (3) as follows:

$$net = \sum_{i=0}^{m} \alpha_i h_i = \mathrm{A.h} \tag{5}$$

where $\alpha \equiv \gamma$. The notations are explained in Fig. 1. The output of FFANN is given as

$$y = f(net) \tag{6}$$

In this paper the linear transformation function has been used in output for (6) and hyperbolic tangent sigmoidal transfer function for hidden layer node in (2).The choice of activation function is based on recommendation in [15] and [16], where author(s) has claimed that better network can be trained with these activation function.

III. Sensitivity Analysis for Node Fault

The sensitivity measures defined herein are based on the variation of the network output and/or network error with the variation in the network hidden nodes' output.

The architecture of the feedforward artificial neural network is illustrated in Fig. 1. Output of the network is defined as:

$$y = \sum_i \alpha_i h_i \left(\sum_i w_{ij} x_j + \theta_i \right) + \gamma \tag{7}$$

Mean Squared Error (MSE) is used to measure the error due to node fault in the Sigmoidal FFANNs. The goodness of the model is measured by the quantity :

$$MSE: \ E = \frac{1}{P} \sum_{p=1}^{P} \left(t^{(p)} - y^{(p)} \right)^2 \tag{8}$$

A. Hidden Node Output Based Sensitivity Measure

In a manner entirely analogous to the establishment of the weight sensitivity measures [17], we may define (where we treat the output bias as a weight associated with an auxiliary hidden node with output value of unity):

$$S_p^{(1)} = \sum_{i=1}^{H} |\, h_i\,|^p + 1 \tag{9}$$

where $p \in R$ and $p \geq 1$. For $p \to \infty$, (9) becomes (after dropping the constant unity term):

$$S_\infty^{(1)} = \max(|\, h_i\,|) \tag{10}$$

where the maximum is over all possible values of *i* ($i \in$(1,2, . . ., H))

B. Gradient With Respect to Hidden Node Output Based Sensitivity Measure

We may define (where we treat the output bias as a weight associated with an auxiliary hidden node with output value of unity):

$$S_p^{(2)} = \sum_{i=1}^{H} |\,\alpha_i\,|^p + |\,\gamma\,|^p \tag{11}$$

where $p \in R$ and $p \geq 1$. For $p \to \infty$, (11) becomes:

$$S_\infty^{(2)} = \max(|\,\alpha_i\,|, |\,\gamma\,|) \tag{12}$$

where the maximum is over all possible values of *i* ($i \in$(1,2, . . ., H)).

C. Hidden Node Output Saliency Based Sensitivity Measure

In a manner entirely analogous to the establishment of the weight saliency [18] based sensitivity measures, we may define (where we treat the output bias as a weight associated with an auxiliary hidden node with output value of unity):

$$S_p^{(3)} = \sum_{i=1}^{H} |\,\alpha_i\,|^p \langle |\, h_i\,|^p \rangle + |\,\gamma\,|^p \tag{13}$$

where $p \in R$ and $p \geq 1$. For $p \to \infty$, (13) becomes:

$$S_\infty^{(3)} = \max(|\,\alpha_i\,|, |\, h_i\,|, |\,\gamma\,|) \tag{14}$$

where the maximum is over all possible values of *i* ($i \in$(1,2, . . ., H)).

D. Condition Number with respect to Hidden Node Output Based Sensitivity Measure

In a manner entirely analogous to the establishment of the condition number for weight-stuck-zero fault [6] & [9] we may define (where we treat the output bias as a weight associated with an auxiliary hidden node with output value of unity):

$$S_{p,1}^{(4)} = \sum_{i=1}^{H} |\,\alpha_i\,|^p \left\langle \left| \frac{h_i}{y} \right|^p \right\rangle + |\,\gamma\,|^p \left\langle \left| \frac{1}{y} \right|^p \right\rangle \tag{15}$$

where $p \in R$ and $p \geq 1$. For $p \to \infty$, (15) becomes:

$$S_{\infty,1}^{(4)} = \max\left(|\alpha_i|, \left| \frac{h_i}{y} \right|, \left| \frac{\gamma}{y} \right| \right) \tag{16}$$

where the maximum is over all possible values of *i* ($i \in$(1,2, . . ., H)).

Using the bounds on the activation function, we obtain:

$$S_{p,2}^{(5)} = \sum_{i=1}^{H} |\,\alpha_i\,|^p \left\langle \left| \frac{1}{y} \right|^p \right\rangle + |\,\gamma\,|^p \left\langle \left| \frac{1}{y} \right|^p \right\rangle \tag{17}$$

where $p \in R$ and $p \geq 1$. For $p \rightarrow \infty$, (17)becomes:

$$S_{\infty,2}^{(5)} = \max\left(\left|\alpha_i\right|, \left|\frac{1}{y}\right|, \left|\frac{\gamma}{y}\right|\right) \tag{18}$$

where the maximum is over all possible values of i ($i \in$(1,2, . . ., H))

E. Node Fault Based Sensitivity Analysis

The modeling of this fault is equivalent to a WSZ fault which occurs only for weights leading out from the hidden node(s). Thus, in a sense this fault is a subset of the weight stuck at zero faults [17] and [20]. For this class of fault, the output of one node at a time was equated to zero, and the fault metric values evaluated for each. The average over all faults, for a network (for a specific task), and patterns is used as the measure of fault effect on the network behaviour.

We analyze the case of a single node stuck at zero, one at a time (the network output is given by relation (7). We consider only the case of a single hidden node bring stuck at zero. Let the *i*th hidden node be stuck at zero, this is equivalent to a WSZ for the α_i weight. Hence, an appropriate measure of the fault is (using (13)):

$$S_{p,1}^{(5)} = \frac{1}{H}\left\{\left\langle \sum_{i=1}^{H} \left|\alpha_i\right|^p \left|h_i\right|^p \right\rangle\right\} \tag{19}$$

Using the bound on the activation function, we obtain another measure of NSZ fault:

$$S_{p,2}^{(5)} = \frac{1}{H}\left\{\sum_{i=1}^{H} \left|\alpha_i\right|^p\right\} \tag{20}$$

IV. Experiments and Results

A small experiment was conducted to demonstrate the applicability of the proposed sensitivity measures of node fault (9) -- (20)) in SFFANNs. Thirty networks were trained for the following approximation task [21].

Fn1: $$y = \sin(x_1 * x_2) \tag{21}$$

x_1,x_2 uniform in [-2,2]

Fn2: $$y = \exp(x_1 * \sin(\pi * x_2)) \tag{22}$$

x_1,x_2 uniform in [-1,1]

Fn3:
$$\begin{aligned} a &= 40 \times \exp(8 \times (x_1 - 0.5)^2 + (x_2 - 0.5)^2) \\ b &= \exp(8 \times ((x_1 - 0.2)^2 + (x_2 - 0.7)^2)) \\ c &= \exp(8 \times ((x_1 - 0.7)^2 + (x_2 - 0.2)^2)) \\ y &= a/(b + c) \end{aligned} \tag{23}$$

x_1,x_2 uniform in [0,1]

Fn4:
$$\begin{aligned} &y = 1.3356[1.5(1 - x_1) + \exp(2x_1 - 1) \\ &\sin(3\pi(x_1 - 0.6)^2) + \exp(3(x_2 - 0.5)) \\ &\sin(4\pi(x_2 - 0.9)^2)] \end{aligned} \tag{24}$$

x_1, x_2 uniform in [0,1]

The data set for ANNs training were generated by uniform sampling of the domain of definition of the functions. The networks consist of two input, one hidden layer and one output node (Fig. 1). The detail of the architecture used is summarized in Table 1. The architecture was identified by exploratory experiments where the size of the hidden layer was varied from 5 to 30 (*i.e.*, the number of nodes in the hidden layer were varied from 5 to 30 in steps of 5) and the architecture that gave the minimum error during training was used. All the hidden node used tangent hyperbolic activation function while the output nodes were linear.

TABLE I: Architecture of Network Used

Sr. No.	Function	Inputs	Hidden nodes	Output Nodes	No. of Weights
1.	Fn1	2	25	1	101
2.	Fn2	2	15	1	61
3.	Fn3	2	20	1	81
4.	Fn4	2	10	1	41

The resilient back propagation (RPROP) [22] algorithm as implemented in MATLAB 7.2 Neural Network toolbox was used with the default learning rate and momentum constant. For training the network, 200 samples were generated from the input domain of the functions. 5000 epochs of training was conducted for each problem. Thirty networks were trained with the above procedure. 200 samples were used for the validation purpose and 19600 samples were used for the testing purpose for each network. Mean squared errors are generated for each network under training, validating and testing routine. A network which has a least validation error is selected for further analysis of proposed sensitivity measures.

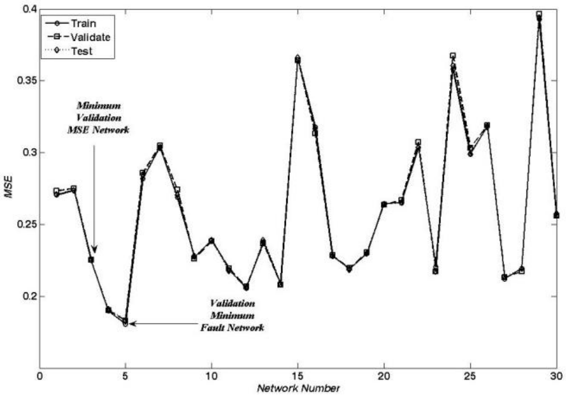

Fig. 2. Demonstration of Error of 30 networks for Fn1

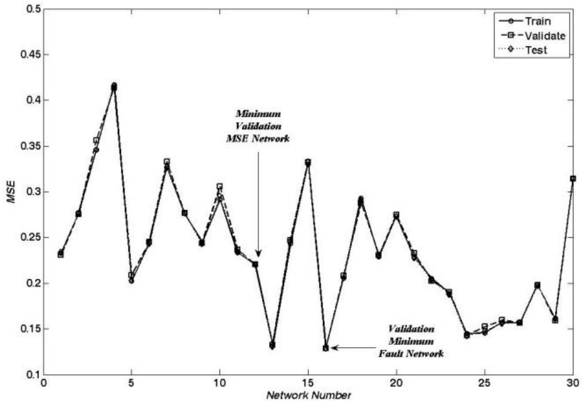

Fig. 3. Demonstration of Error of 30 networks for Fn2

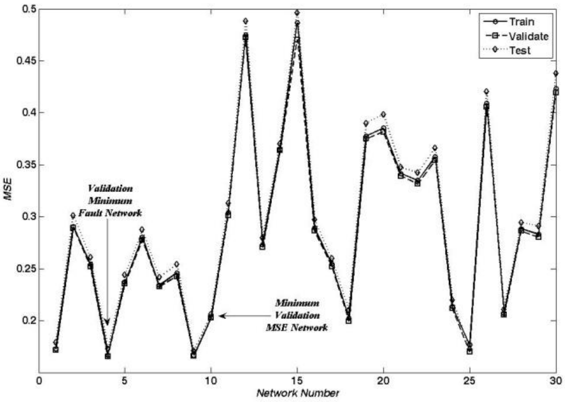

Fig. 4. Demonstration of Error of 30 networks for Fn3

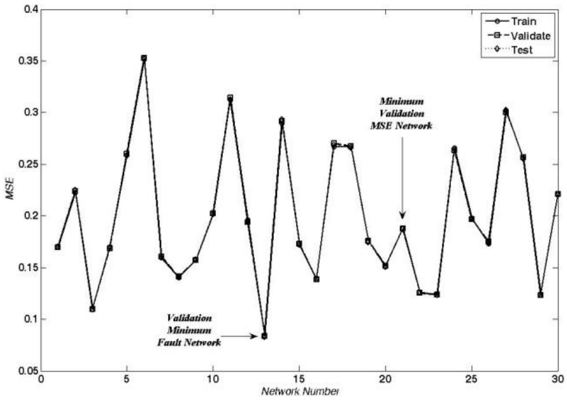

Fig. 5. Demonstration of Error of 30 networks for Fn4

TABLE II: CORRELATION COEFFICIENT

Sr. No.	Function	$S_2^1(9)$	$S_2^2(11)$	$S_2^3(13)$	$S_{2,1}^4(15)$	$S_{2,2}^4(17)$	$S_{2,1}^5(19)$	$S_{2,2}^5(20)$
1.	Fn1	-0.01(0.97)	0.97(0.00)	0.98(0.00)	-0.14(0.45)	-0.20(0.01)	0.79(0.00)	0.97(0.00)
2.	Fn2	-0.18(0.35)	0.98(0.00)	0.98(0.00)	-0.12(0.54)	0.36(0.00)	0.86(0.00)	0.99(0.00)
3.	Fn3	0.14(0.45)	0.92(0.00)	0.93(0.00)	0.45(0.01)	0.30(0.04)	0.86(0.00)	0.96(0.00)
4.	Fn4	-0.08(0.68)	0.94(0.00)	0.98(0.00)	0.30(0.10)	-0.11(0.01)	0.91(0.00)	0.95(0.00)

For this fault, the network with the minimum validation error need not be the network that gives the minimum error under fault. This is illustrated in Fig. 2 - Fig. 5 (for Fn1 - Fn4, respectively). The network with the minimum validation MSE and the network with the minimum value of the fault metric value are indicated. Fig. 2 - Fig. 5 demonstrate that in general the minimum validation error network and the minimum fault metric value networks differ for specific tasks.

From these figures it is clear that the network with the minimum validation error does not necessarily have the best behavior under the fault. Another striking facet of these figures is the observation that irrespective of whether we consider the fault measure value over the training set or the validation / test set, the fault metric values appear to be very close. This lead us to conjecture that -- the fault metric value for the train set, the validation set and the test set all belong to the same distribution. The result of the *t*-test [23] and the Kolmogrov-Smirnov test [24] performed for the verification of this conjecture showed that the fault metric value for these three dataset belonged to the same distribution at the 5% significance level.

This shows that for at least the NSZ analysis, the data over the training set or the validation set (that is the data usually available before commencement of training) is sufficient for the fault tolerance studies. Thus, for NSZ fault analysis in this paper validation data set is used. (Though we could have used the training dataset also, the choice is arbitrary). This is a significant result in the sense that the behavior on unknown data sets can be figured out in terms of the available data set.

The coefficient of correlation values between the fault measures (8), NSZ fault empirical MSE and the sensitivity measures proposed in this paper for NSZ fault is shown in Table II. The figure in brackets represents the probability of obtaining the correlation as large as the observed value by random chance, when the true correlation is zero. If probability is small (less than 0.05), then the correlation is significant.

V. CONCLUSION

In this paper, detailed experiments are performed for four approximation tasks. Validation error is used for the selection of best network for further analysis. Chosen network was used for further analysis of node stuck-at-zero fault. Experimental result demonstrates that $S_2^2(11), S_2^3(13), S_{2,1}^5(19)$ and $S_{2,2}^5(20)$ statistically significant measures of the NSZ fault at 95% confidence level and correlation value at 0.75.

REFERENCES

[1] D. S. Phatak and I. Koren, "Complete and partial fault tolerance of feedfoward neural netws," *IEEE Transaction on Neural Networks,* vol. 6, no. 2, pp. 446-456, 1995.

[2] G. Bolt, "Investigatiing fault tolerance in artificial neural networks," Department of computer Science, University of York, Hesington, York, England, Tech. Rep. YCS, vol. 154, 1991.

[3] P. Chandra and Y. Singh, "Feed forward sigmoidal networks-equicontinuinuity and fault-tolerance properties," *IEEE Transaction on Neural Networks,* vol. 15, no. 6, pp. 435-444, 2004.

[4] F. M. Dias and A. Antunes, "Fault tolerance of artificial neural networks: an open discussion for a global model," *International Journal of Circuits, Systems and Signal Processing,* vol. 4, no. 1, pp. 9-16, 2010.

[5] P. Chandra and Y. Singh, "Fault tolerance of feedforward artificial neural networks- a framework of study," in *IEEE Joint Conference on Neural Networks,* 2003, pp. 489-494.

[6] A. P. Singh, P. Chandra, and C. S. Rai, "Fault models for neural hardware," in *Proc. IEEE First International Conference on Advances in System Testing and Validation Lifecycle (VALID 2009),* Port, Portugal, 2009, pp. 7-12.

[7] M. Minnix, "Fault tolerance of the backpropation neural network trained on noisy inputs," in *IEEE International Joint Conference on Neural Networks,* vol. 1, 1992, pp. I-847-852.

[8] B. Segee and M. Carter, "Fault sensitivity and nodal relevance relationship in multilayer perceptrons," Intelligent System Group, robotics Laboratory, electrical and Computer Engineering Department: Durham, University of New Hampshire, Tech. Rep. ECE.IS.90.0,1990

[9] A. P. Singh, C. S. Rai, and P. Chandra, "Empirical study of FFANN tolerance to weight stuck at MAX/MIN fault," *International Journal of Artificial Itelligence & Applications,* vol. 1, no. 2, pp. 13-21, 2010.

[10] A. F. Murray and P. J. Edwards, "Synaptic weight noise during MLP training: Enhanced MLP performance and fault tolerance resulting from synaptic weight noise during training," *IEEE Transaction of Neural Networks,* vol. 5, no. 5, pp. 792-802, 1994.

[11] J. Sum, A. C. S. Leung, and K. Ho, "Prediction error of a fault tolerant neural network," *LNCS Neural Information Processing,* vol. 4232, no. 521, pp. 107-110, 2006.

[12] J. Sum and A. C. S. Leung, "Prediction error of a fault tolerant neural network," *Neurocomputing*, 2008.

[13] B. Sher and W. Hsieh, "Fault tolerance training of feedfoward neural networks," in *Proc. the National Science Council, Republic of China*, vol. 23, no. 5, pp. 599-608, 1999.

[14] Z. H. Zouh, S. F. Chen, and Z. Q. Chen, "Improving tolerance of neural networks against multi-node open faults," in *Proc. IJCNN'01,* vol. III, 2001, pp. 1687-1692.

[15] T. Vogl, J. Mangis, A. Rigler, W. Zink, and D. Alkon, "Accelerating the convergence of the backpropagation method," *Biological Cybernetics*, vol. 59, pp. 257-263, 1988.

[16] Y. Lecun, L. Bottou, G. Orr, and K. Muller, "Efficient backprop," *Neural Networks: Tricks of the trade. Springer,* 1998.

[17] A. P. Singh, P. Chandra, and C. S. Rai, "New Metric for sensitivity analysis of FFANN," *International Journal of Engineering Science & Technology*, vol. 2, no. 4, pp. 322-329, 2010.

[18] M. C. Mozer and P. Smolensky, "Skeletonization: A technique for trimming the fat from a network via relevance assessment," in *Advances in Neural Information Processing Systems,* D. S. Touretzky, Ed. Morgan Kaufmann, 1989, vol. 1, pp. 107-115.

[19] S. C. Chapra and R. P. Canale, *Numerical Methods for Engineers.* New York: McGraw Hill, 2002.

[20] A. P. Singh, P. Chandra, and C. S. Rai, "Empirical study of FFANNs tolerance to weight stuck at zero fault," *International Journal of Engineering & Technology,* vol. 2, no. 2, pp. 65-70, 2010.

[21] V. Cherkassky, "Comparison of adaptive methods for function estimation from samples," *IEEE Transaction on Neural Networks,* vol. 7, no. 4, pp. 969-984, 1996.

[22] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropgation learning: The RPROP algorithm," in *Proc. the IEEE International Conference on Neural networks (ICNN),* San Francisco, 1993, pp. 586-591.

[23] F. J. Massey, "The Kolmogorov-Smirnov test for goodnes of fit," *Journal of the American Statistical Association,* vol. 46, no. 253, pp. 68-78, 1951.

[24] L. H. Miller, "Table of percentage points of Kolmogorov statistics," *Journal of the American Statistical Association,* vol. 51, no. 273, pp. 111-121, 1956.

Amit Prakash Singh is an associate professor in University School of Information & Communication Technology, Guru Gobind Singh Indraprastha University, Delhi. He did Ph.D. in Information Technology from GGS Indraprastha University, Delhi. His area of research is Artificial Neural Network, Embedded System and Digital System Design. He is a member of IACSIT, IEEE, Computer Society of India, VLSI society of India and IETE.

Chandra Shekhar Rai is a professor in University School of Information & Communication Technology, Guru Gobind Singh Indraprastha University, Delhi. He has 14 years of teaching and research experience. He has published many research papers in the journal of international repute. His area of research is artificial neural network and signal processing.

Pravin Chandra is a professor in University School of Information & Communication Technology, Guru Gobind Singh Indraprastha University, Delhi. He has also worked in University of Delhi for one year. He has published many research papers in the area of Artificial Neural Network in journal of international repute. He is a IEEE member.