IEICE TRANS. INF. & SYST., VOL. E81–D, NO. 1 JANUARY 1998

PAPER

On the Activation Function and Fault Tolerance in Feedforward Neural Networks

Nait Charif HAMMADI† ***and*** **Hideo ITO**††, ***Members***

SUMMARY Considering the pattern classification/recognition tasks, the influence of the activation function on fault tolerance property of feedforward neural networks is empirically investigated. The simulation results show that the activation function largely influences the fault tolerance and the generalization property of neural networks. It is found that, neural networks with symmetric sigmoid activation function are largely fault tolerant than the networks with asymmetric sigmoid function. However the close relation between the fault tolerance and the generalization property was not observed and the networks with asymmetric activation function slightly generalize better than the networks with the symmetric activation function. First, the influence of the activation function on fault tolerance property of neural networks is investigated on the XOR problem, then the results are generalized by evaluating the fault tolerance property of different NNs implementing different benchmark problems.

key words: *feedforward neural network, XOR problem, critical weights, fault tolerance, generalization ability*

1. Introduction

Feedforward neural networks (NNs), trained with backpropagation algorithm (BP) have been applied successfully in variety of diverse area such as speech recognition, optical character recognition, control and medical analysis [1]. The BP algorithm seeks to minimize the error in the output of the network as compared to a target, or desired response [2]. Although it was thought that NNs are fault tolerant as they consist of parallel processing elements, recently extensive research has proved that NNs are not intrinsically fault tolerant, and the fault tolerance has to be enhanced by adequate scheme [3],[4].

The influence of learning time, training with noisy input data, and noise injection on the synaptic weights during training on the neural networks under the existence of fault has been studied [3],[5],[6]. The ability to perform the generalization is one of the most important properties of NNs [11]. In an attempt to improve the generalization capability, a weight smoothing algorithm was proposed in [12]. This algorithm can be used when there exists some correlations among neighboring data patterns. The pruning techniques were also proposed to enhance the generalization cability of the NNs [10],[15].

Manuscript received August 23, 1996.
Manuscript revised July 10, 1997.

†The author is with the Graduate School of Science and Technology, Chiba University, Chiba-shi, 263 Japan.
††The author is with the Faculty of Engineering, Chiba University, Chiba-shi, 263 Japan.

However, the influence of the activation function, which is an important element in neural networks design, on the fault tolerance and on the generalization ability of the NNs has not yet been investigated.

In the design of NNs one has to choose parameters like the network topology, the learning rate, the initial weights, and the activation function. In this paper the influence of the activation function on the fault tolerance and the generalization ability of feedforward neural networks are investigated.

It should be noted that the fault tolerance property, as the ability to function in the presence of faults, is different from the ability to classify non-trained data (generalization ability). Therefore, the training set is used to assess the fault tolerance property and the set of non-trained patterns is used to assess the generalization ability.

Next section describes the neural network architecture used in the experiments and presents the fault model adopted. In Sect. 3 the influence of the activation function on fault tolerance property of NN trained on the XOR problem is presented and discussed. Section 4 evaluates the result on a collection of benchmak problems. In Sect. 5 we investigate the influence of the activation function on the generalization property of NNs.

2. Neural Network Architecture and Fault Model

2.1 Neural Network Architecture

We consider a feedforward neural network with single hidden layer denoted by $I-H-K$, where I, H and K are the number of neurons in the input layer, the hidden layer, and the output layer respectively. The input layer neurons are fully connected to the hidden layer neurons which are fully connected to the output layer neurons. The output of the ith neuron is given by

$$o_i = f\left(\sum_{j=0}^{N_i} w_{ji} o_j\right), \tag{1}$$

where w_{ji} is the synaptic weight corresponding to the connection from the jth neuron to the ith neuron, N_i is the number of neurons that feeds the ith neuron (which is equal to the number of neurons in the previous layer)

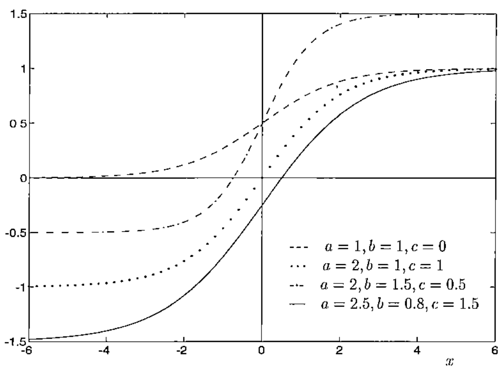


Fig. 1 Examples of activation functions.

and f is an activation function. The backpropagation algorithm requires a continuous differentiable activation function. A commonly used activation is a sigmoid function given by

$$f(x) = \frac{a}{1 + e^{-bx}} - c, \tag{2}$$

where a, b and c are real positive numbers [14]. The activation function can be symmetric or asymmetric depending on the values of a and c. The popular examples of sigmoid function are, the hyperbolic tangent defined by ($a = 2$, $b = 1$, and $c = 1$) giving symmetric output $(-1, 1)$, and the asymmetric function defined by ($a = 2$, $b = 1$, $c = 0$) giving an output (0,1). A symmetric function defined by ($a = 1$, $b = 1$, $c = 0.5$) has been also used in [14]. Figure 1 shows some examples of the activation function.

2.2 Input and Output Scaling

The network outputs vary between $-c$ and $a - c$, thus the target values are rescaled to make their range in $[-c, a - c]$ interval [18]. Such restriction has no influence on the performance of networks to solve pattern classification problem [19]. From the theoritical point of view the scaling of the input is not necessary. However large input values may result in the deep saturation of the hidden neurons. To avoid such situation, the inputs are scaled in the same range as the output [18]. In this paper, we adopt the linear scaling given by the following equation

$$\lambda_i^{p*} = a\frac{\lambda_i^p - \lambda_{i\,\min}^p}{\lambda_{i\,\max}^p - \lambda_{i\,\min}^p} - c, \tag{3}$$

where the element λ_i^p the ith element of the p^{th} sample rescaled to λ_i^{p*}, $\lambda_{i\,\min}^p = \min_p\{\lambda_i^p\}$ and $\lambda_{i\,\max}^p = \operatorname{Max}_p\{\lambda_i^p\}$.

2.3 Fault Model

A physically plausible type of fault is the loss of connection between two neurons (*open fault*) [8], this relates to the loss of an arc in a directed graph which abstractly represents the topology of NNs [13]. This fault is equivalent to the case when the synaptic weight is set at 0, which is equivalent to the conventional stuck-at-0 type. In addition, the conventional stuck-at-1 fault of synaptic weights is also considered.

Fault tolerance is frequently cited as an important property of NNs [8], however, a single fault is frequently sufficient to completely disrupt a learned function. A synaptic weight is called a critical weight for a given type of fault if its stuck-at-fault causes an error at the output [8]. Ncw_0 is defined as the number of critical weights in NN when the single stuck-at-0 fault is assumed, and Ncw_1 the number of critical weights when the single stuck-at-1 fault is assumed. In the simulation, to find the number of critical weights Ncw_0 (Ncw_1), a synaptic weight is set at 0 (at 1) and all training patterns are presented to NN. If the network could not recognize all the training patterns the given link is a critical one. The sum of critical weights $SNcw = Ncw_0 + Ncw_1$ measures the ability of the network to maintain function in the presence of a single fault. The network is completely fault tolerant when the $SNcw$ is zero.

The output of the neuron k in the output layer is classified as follows:

$$\begin{aligned} o_k &= a - c \quad \text{if} \quad \sigma = \sum_{j=0}^{N_k} w_{jk} o_j > 0, \\ o_k &= -c \quad \text{if} \quad \sigma = \sum_{j=0}^{N_k} w_{jk} o_j < 0. \end{aligned} \tag{4}$$

The output is considered wrong if it switches from $a - c$ to $-c$ or from $-c$ to $a - c$, this happens if the sign of σ changes.

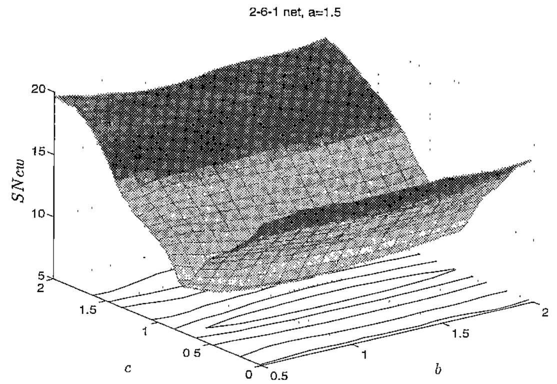
3. Activation Function on XOR Problem

The XOR function, which may be viewed as a special case of more general nonlinear problems, is considered to investigate the influence of the activation function on the fault tolerance ability of feedforward neural networks. The results are generalized by evaluating the fault tolerance property of different NNs implementing different benchmark problems (Sects. 4 and 5).

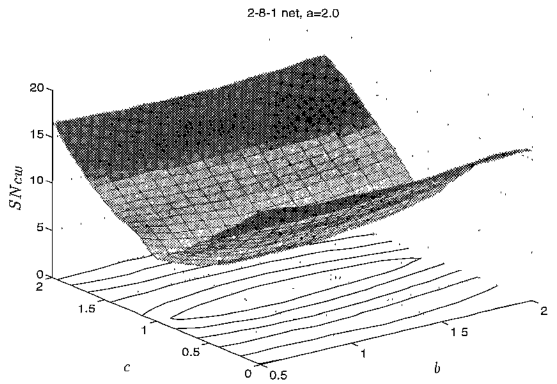
The symmetric sigmoid functions or hyperbolic tangent ($a = 2$, $b = 1$, $c = 1$) and the asymmetric one ($a = 2$, $b = 1$, $c = 0$) are probably the most often applied activation functions in feedforward neural networks [15]. In this section the influence of the parameters a, b and c on the fault tolerance property of feedforward neural network is investigated.

3.1 Single Fault

The sum of critical weights $SNcw = Ncw_0 + Ncw_1$ is the fault tolerance metric adopted for the XOR problem



(a) $a = 1.5$ for the 2-6-1 NN



(b) $a = 2$ for the 2-8-1 NN.

Fig. 2 The surface graph and the corresponding contour plot representing $SNcw$, in the NNs with sigmoid activation function, as a function of b and c.

in the case of single fault. The $SNcw$ will be assessed as function of a, b and c.

The performance of networks depends on the learning rate [3], the random start position (initial weights values) of the network [9],[10], and the size of the network. In this paper to evaluate the fault tolerance of a given network with a given activation function (i.e., given a, b and c), the learning rate is fixed at 0.1, and 200 random start positions are used and the results are averaged.

For each triple a, b, and c, the neural network is trained with the backpropagation algorithm until it recognize all training patterns. Then it is tested and the number of critical weights is calculated. The sum of critical weights $SNcw$ is a multivariable function that can not be plotted in the two dimensional graph. Figure 2 depicts the influence of the parameters b and c on the fault tolerance of the network when a is constant. It shows the surface graph with the corresponding contour plot (i.e., level lines) of $SNcw$. The vertical axis presents the sum of critical weights $SNcw$. Independently of the number of hidden neurons and the value of a, all the graphs have a similar shape and the sum of critical weights $SNcw$ is minimal for $c = a/2$, which is a symmetric function.

It can be realized from the simulation results shown in Fig. 2 that, NNs with symmetric activation function have significantly less number of critical weights than the NNs with asymmetric function. This means that the probability to get a wrong output, when a single fault is present, is greater for NNs with asymmetric activation function than the NNs with symmetric activation function.

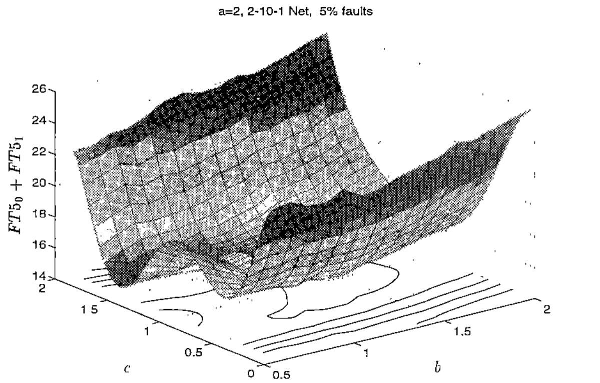
3.2 Multiple Faults

The presence of multiple faults in the network is assessed by setting a number of randomly selected weights to faults. For convenience, the ability of NNs to tolerate a multiple stuck-at-0 faults and the ability to tolerate a multiple stuck-at-1 faults are assessed separately. The patterns of the training set are applied, and the percentage of recognized patterns is assessed for each number of faulty weights. As some weights are more significant than others, the process was repeated 200 times and the results are averaged. As the results depend on the random start position of the network [9],[10], two handred random start position are used for each neural network and the results are averaged. In this case of multiple faults, the FT is measured in percent of incorrectly classified/recognized patterns as function of b and c when a given persecentage of weights is faulty. We define the following variables, $FT5_0$ (respectively $FT10_0$) the percentage of incorrectly classified/recognized patterns when 5% of the weights (respectively 10%) are stuck-at-0 faults, and $FT5_1$ (respectively $FT10_1$) the percentage of incorrectly classified/recognized patterns when 5% of the weights (respectively 10%) are stuck-at-1 faults. Although many networks were investigated with different number of multiple faults, only two graphs are presented here. Figure 3 depicts the influence of the parameters b and c on the fault tolerance of the network when a is constant. It shows the surface graph with the corresponding contour plot (i.e., level lines) of $FT5_0 + FT5_1$ and $FT10_0 + FT10_1$. The vertical axis presents the sum of incorrectly classified patterns $FT5_0 + FT5_1$ and $FT10_0 + FT10_1$. Independently of the number of hidden neurons and the value of a, all the graphs have a similar shape.

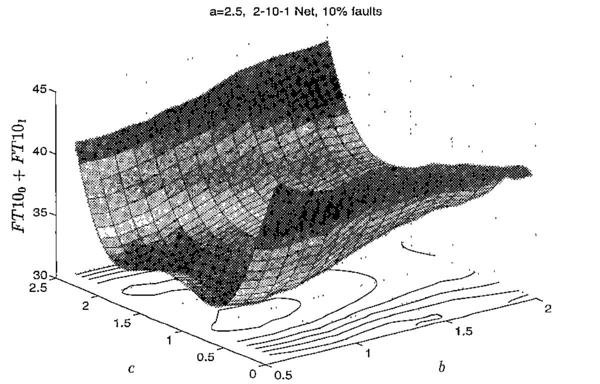
From Fig. 3 and Fig. 4 it can be realzed that the networks with symetric activation function have quite better fault tolerance property than the networks with assymetric activation function. Although the minimum of $FT5_0 + FT5_1$ and $FT10_0 + FT10_1$ are not obtained exacltly for symetric activation function.

4. Result Evaluation

In this section the obtained results above are generalized by evaluating the fault tolerance property of different NNs implementing different benchmark problems. Two



(a)$a = 2$, 5% of faulty weights



(b)$a = 2.5$, 10% of faulty weights NN.

Fig. 3 The surface graph and the corresponding contour plot representing the % of incorrectly patterns by 2-10-1 network when 5% (respectively 10%) of its weights are faulty, as a function of b and c.

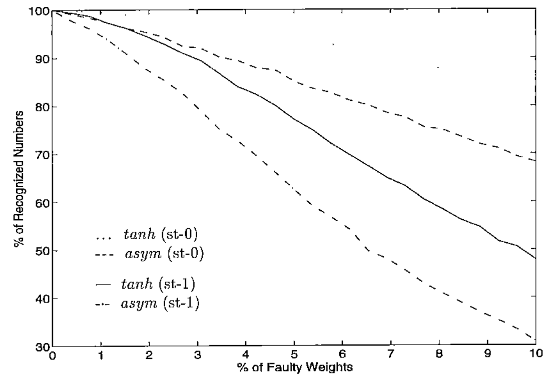


Fig. 4 The percentage of recognized numbers by 49-10-10 networks as function of the % of faulty weights stuck-at-0/stuck-at-1.

Table 1 Number of critical weights Ncw_0 in numbers recognition problem.

Networks	42-6-10	42-10-10	42-14-10	42-18-10
$tanhNcw_0$	27.7	3.2	0.3	0.0
$asymNcw_0$	40.4	16.5	6.1	2.7
$tanhNcw_1$	35.4	6.7	2.3	0.6
$asymNcw_1$	40.8	26.2	18.1	11.4

types of NNs are investigated, the first one has the hyperbolic tangent activation function (denoted by $tanh$) ($a = 2, b = 1, c = 1$) which is the commun symmetric function, and the second one ($asym$) has asymmetric activation function ($a = 1, b = 1, c = 0$).

4.1 Numbers Recognition Problem

The numbers from 0 to 9 presented on 7×6 image plane are used to evaluate the neural network's ability to tolerate faults. This set is fairly difficult for classification since the pairs of patterns (particularly (6,8), and (8,9)) are close to each other in the input space (i.e., the Hamming distance is only 2). The normalized patterns from 0 to 9 are used in both training and testing phases. Networks with 6, 10, 14, and 18 hidden neurons are invetigated. Each network is trained by the backpropagation algorithm. After the training has been finished, the network's tolerance to damage is assessed. Twenty experiments are made for each network with different initial weights.

Table 1 shows the number of critical weights Ncw_0 and Ncw_1, in the networks. Figure 4 presents the percentage of recognized patterns by 49-10-10 network as function of the % of faulty weights (that are stuck-at-0 or stuck-at-1).

It can be realized from Table 1 and Fig. 4 that the neural networks with symmetric sigmoid function exhibit better graceful degradation than the networks with asymmetric function.

4.2 Parity Check Problem

In this section a parity check problem is considered in which the input patterns are six dimensional. The desired output is 0 if the corresponding input is made up of odd number of 1, otherwise the desired output is set to 0. Networks with 8, 12, 16, and 20 neurons are investigated.

Table 2 shows the number of critical weights Ncw_0 and Ncw_1. The results show that the number of critical weights in the networks with symmetric sigmoid function $tanh$ is less than in the networks with asymmetric function $asym$. Figure 5 presents the percentage of correctly checked parity using 6-20-1 network as function of the number of faults.

It can be realized from Table 2 and Fig. 5 that the neural networks with symmetric sigmoid function exhibit better graceful degradation than the networks with asymmetric function.

4.3 Contact Lens Prescription

The data for the contact lens prescription was obtained from ftp.ics.uci.edu/pub/machine-learning-databases/. Based on the the age of a patient, his spectacle prescrip-

Table 2 Number of critical weights Ncw_0 and Ncw_1 for parity check problem.

Networks	6-8-1	6-12-1	6-16-1	6-20-1
$tanh Ncw_0$	45.3	36.7	25.9	16.8
$asym Ncw_0$	53.5	60.7	62.3	62.7
$tanh Ncw_1$	38.4	36.6	32.6	26.0
$asym Ncw_1$	51.6	61.0	59.4	64.4

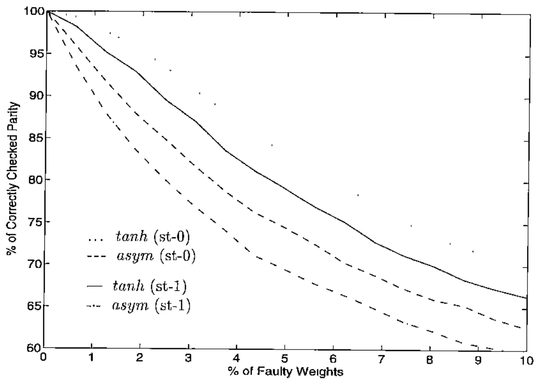


Fig. 5 The percentage of correctly checked parity by 6-20-1 networks as function of the % of faulty weights.

Table 3 Number of critical weights Ncw_0 and Ncw_1 for contact lens prescription problem.

Networks	4-2-3	4-4-3	4-6-3	4-8-3
$tanh\ Ncw_0$	15.5	15.2	11.9	8.2
$asym\ Ncw_0$	17.4	22.9	26.2	28.4
$tanh\ Ncw_1$	15.7	15.3	14.1	12.7
$asym\ Ncw_1$	17.6	22.5	25.7	27.5

tion, the tear production rate, and whether he is astigmatic or not, it should be decieded which type contact lenses he should be fitted (soft or hard), or he should not be fitted with any one (no contact lens). This data base consists of 24 examples, each example is a four dimentional vector. Network 4-2-3, 4-4-3, 4-6-3, and 4-8-3 are investigated.

Table 3 shows the number of critical weights Ncw_0 and Ncw_1. Figure 6 presents the percentage of correctly taken decisions as function of the percentage of faulty weights.

It can be also realized from Table 3 and Fig. 6 that neural networks with symmetric sigmoid function exhibit better graceful degradation than the networks with asymmetric function.

4.4 Iris Problem

The Iris problem is a well-known classification problem. It consists of classification of 150 samples in 3 classes. The data were obtained from the same location as the "Contact Lens Prescription." In our experiment the training test of size 100 consists of the first 33, 33, and 34 samples of the three classes, respectively. The remaining 50 samples represent the test set. Network 4-

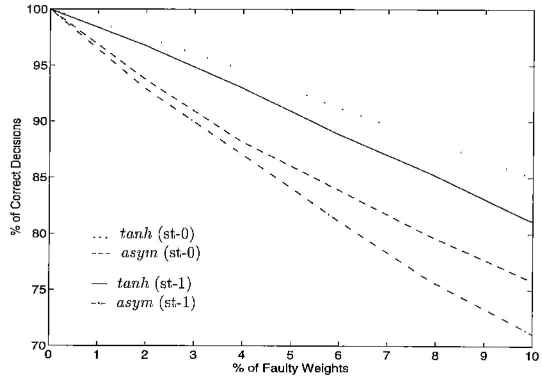


Fig. 6 The percentage of correctly taken decisions by 4-6-3 networks as function of the % of faulty weights.

Table 4 Number of critical weights Ncw_0 and Ncw_1 for Iris problem.

Networks	4-4-3	4-8-3	4-12-3	4-14-3
$tanh\ Ncw_0$	12.2	10.2	9.6	8.7
$asym\ Ncw_0$	18.5	23.5	27.7	34.6
$tanh\ Ncw_1$	11.8	10.8	10.2	10.1
$asym\ Ncw_1$	18.7	24.9	29.5	38.3

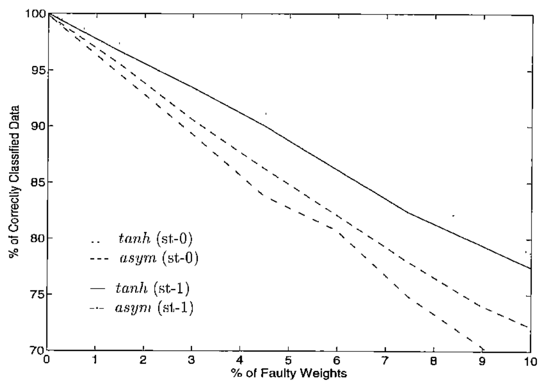


Fig. 7 The percentage of correctly classified samples by 4-8-3 networks as function of the % of faulty weights.

4-3, 4-8-3, 4-12-3, and 4-16-3 are traind to calassify the data.

Table 4 shows the number of critical weights Ncw_0 and Ncw_1. Figure 7 presents the percentage of correctly classified samples as function of the number of faults by 4-8-3 network.

It can be realized from Table 4 and Fig. 7 that the neural networks with symmetric sigmoid function exhibit better graceful degradation than the networks with asymmetric function.

4.5 Mechanical Parts Classification

This problem consists of classification of mechanical parts into seven classes based on simularity feature[17]. The training set consist of 19 mechanical parts, each part is presented on 6×9 pixels.

Table 5 Number of critical weights Ncw_0 and Ncw_0 in the case of Mechanical Parts Classification problem.

Networks	54-4-3	54-6-3	54-8-3	54-10-3
tanh	3.0	0.9	0.2	0.0
asym	5.6	7.3	4.2	3.5
tanh	3.1	1.3	0.7	0.0
asym	8.9	7.4	6.6	5.4

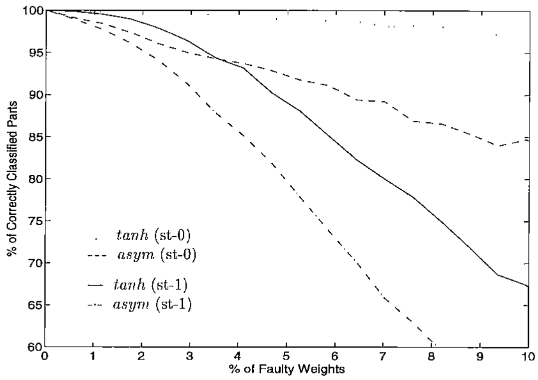


Fig. 8 The percentage of recognized parts by 54-8-3 networks as function of the % of faulty weights.

Table 5 show the number of critical weights Ncw_0 and Ncw_1. Figure 8 presents the percentage of correctly classified samples as function of the number of faults using 54-8-3 network. It can be realized from Table 5 and Fig. 8 that the neural networks with symmetric sigmoid function exhibit better graceful degradation than the networks with asymmetric function.

5. Generalization Ability

Since it is usually hard in practical application to get sufficient amount of training due to finite amount of data and training time, one can expect that network will be able to classify non-trained patterns by applying the generalization principle [3].

5.1 Number Recognition Problem

To evaluate the generalization ability of NNs with the activation functions *tanh*, and *asym*, a test set of 4000 patterns was generated by changing 1, 2, and 3 randomly selected pixels in each normal pattern (400 test patterns from each number). The patterns of the test set are applied to the networks, and the percentage of recognized patterns is assessed. Table 6 shows the results.

Examining Table 6, which presents the average recognation rates on the test set, it can be seen that the NNs with asymmetric activation function has slightly better generalization ability than the NNs with symmetric activation function.

5.2 Mechanical Parts Classification

A test set of 3800 patterns was generated by changing 1, 2, 3, 4, and 5 randomly selected pixels in each normal pattern (200 test patterns of each pattern). The patterns of the test set are applied to the networks, and the percentage of recognized patterns is assessed. Table 4 shows the results.

Table 6 The % of recognized patterns of test set.

Net	42-6-10	42-10-10	42-14-10	42-18-10
tanh	81.4	92.5	93.9	94.0
asym	90.8	93.7	94.4	94.2

Table 7 The % of recognized patterns of test set.

Net	54-4-7	54-6-7	54-8-7	54-10-7
tanh	95.04	96.36	96.42	96.78
asym	97.19	98.49	98.58	98.92

Examining Table 7, it can be seen that the NNs with asymmetric activation function has slightly better generalization ability than the NNs with symmetric activation function.

For the Iris Problem, all networks were able to classify 98% of the test set. This means that the generalization ability was the same regardless of the activation function in the case of Iris data.

6. Conclusion

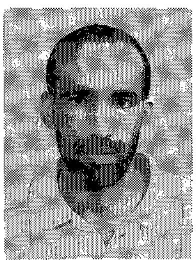
The influence of the activation function on the fault tolerance property of feedforward neural network trained with the backpropagation algorithm have been investigated. It is found that the NNs with symmetric sigmoid function have better fault tolerance property than the NNs with asymmetric function. It was also found that the networks with asymmetric activation function slightly generalize better than the NNs with symmetric activation.

The result obtained in this papaer are expected to help the network designers to choose the activation function that yields to networks with better fault tolerance property.

References

[1] S. Hayken, "Neural Networks, A comprehensive foundation," IEEE Press, 1994.
[2] R.C. Frye, E.A. Rietman, and C.C. Wong, "Backpropagation learning and nonidealities in analog neural network hardware," IEEE Trans. Neural Networks, vol.2, no.1, pp.110–117, Jan. 1991.
[3] J. Nijhuis, B. Hofflinger, A. Schaik, and L. Spaanenburg, "Limits to fault-tolerance of a feedforward neural network with learning," Digest of Fault Tolerant Computing Symposium, pp.228–235, June 1990.
[4] B.E. Segee and M.J. Carter, "Fault tolerance of pruned multilayer networks," Digest of Int. Joint Conf. Neural Networks, July 1991.
[5] J.I. Minnix, "Fault tolerance of backpropagation neural network trained on noisy inputs," Digest of Int. Joint Conf. Neural Networks, June 1992.
[6] A.F. Muroay and P.J. Edwards, "Enhanced MLP perfor-

mance and fault tolerance resulting from synaptic weight noise during training," IEEE Trans. Neural Networks, vol.5, no.5, pp.792–802, 1994.
[7] M.D. Emmerson and R.I. Damper, "Determining and improving the fault tolerance of multi layer perception in a pattern-recognition application," IEEE Trans. Neural Networks, vol.4, no.5, pp.788–793, 1993.
[8] Y. Tan and T. Nanya, "Fault-tolerant back-propagation model and its generalization ability," Digest of Int. Joint Conf. Neural Networks, 1993.
[9] G.P. Drago and S. Ridella, "Statistically controlled activation weight initialization (SCAWI)," IEEE Trans. Neural Networks, vol.3, no.4, pp.627–631, July 1992.
[10] R. Reed, "Pruning algorithms — A survey," IEEE Trans. Neural Networks, vol.4, no.5, pp.740–747, Sept. 1993.
[11] S.B. Holden and P.J.W. Rayner, "Generalization and PAC learning: Some new results for the class of generalized single-layer networks," IEEE Trans. Neural Networks, vol.6, no.2, pp.468–480, March 1995.
[12] J.S.N. Jean and J. Wang, "Weight smoothing to improve network generalization," IEEE Trans. Neural Networks, vol.5, no.5, pp.468–480, Sept. 1995.
[13] G. Bolt, "Fault models for artificial neural networks," Digest of Int. Joint Conf. Neural Networks, pp.1373–1378, 1991.
[14] D.S. Phatak and I. Koren, "Complete and partial fault tolerance of feedforward neural nets," IEEE Trans. Neural Networks, vol.6, no.2, pp.446–456, March 1995.
[15] E.D. Karnon, "A simple procedure for pruning back-propagation trained neural networks," IEEE Trans. Neural Networks, vol.1, no.2, pp.239–242, June 1995.
[16] X.-H. Yu, G.-A. Chen, and S.-X. Cheng, "Dynamic learning rate optimization of the backpropagation algorithm," IEEE Trans. Neural Networks, vol.6, no.3, pp.669–677, May 1995.
[17] N.S. Merchawi, S.T. Kumara, and C.R. Das, "A probabilistic model for the fault tolerance of multilayer perceptrons," IEEE Trans. Neural Networks, vol.7, no.1, pp.201–205, Jan. 1996.
[18] L.P.J. Veelenturf, "Analysis and Applications of Artificial Neural Networks," Prentice Hall International (UK) Ltd., 1995.
[19] J. Cao and A. Cichocki, "Adaptive multilayer perceptron with automatic gain scaling," IEICE Technical Report, NC96-117, June 1996.



Nait Charif Hammadi was born in Tinghir, Warzazat, Morocco, on December 25, 1965. He received the Ingenieur d'Etat diploma in electronics and control, from Ecole Hassania des Traveaux Publics, Casablanca, Morocco, in 1990. In 1991, he joined the Ecole Superieure de Technologie, Mohamed I University, as lecturer. Currently, he is working toward his Ph.D. in Information and Computer Sciences, Graduate School of Science and Technology, Chiba university, Japan. His research interests include neural networks and fault tolerant systems. He is a member of IEEE.

Hideo Ito was born in Chiba, Japan, on June 1, 1946. He received the B.E. degree from Chiba University in 1969, and the D.E. degree from Tokyo Institute of Technology in 1984. He joined Nippon Electric Co. Ltd. in 1969, and Kisarazu Technical College in 1971. Since 1973 he has been a member of the Chiba University. He is currently a Professor of Department of Information and Computer Sciences. His research interests include easily testable design, test generation, VLSI architecture, fault-tolerant design of parallel & distributed systems, and defect-tolerant VLSI design. He is a member of the Information Processing Society of Japan and the IEEE Computer Society.