

## STABILITY PREDICTION OF BERM BREAKWATER USING NEURAL NETWORK

by

S. Mandal<sup>1</sup>, Subba Rao<sup>2</sup> and Y.R. Manjunath<sup>3</sup>

### ABSTRACT

In the present study, an artificial neural network method has been applied to predict the stability of berm breakwaters. Four neural network models are constructed based on the parameters which influence the stability of breakwater. Training and testing of the network models are carried out for different hidden nodes and epochs. The network models' results are compared with the empirical relationships. It is observed that the correlation between predicted stability values by the network models and estimated values are in good agreement.

### 1. INTRODUCTION

The stability of a breakwater is most widely analyzed using well-known formulae given by Hudson (1958) and Van der Meer (1988). The armour weight is calculated using Hudson equation. Although these formulae are formulated based on the many experimental results, still they show disagreement between each other. Therefore, a number of studies have been carried out further to develop new empirical formulae for breakwater stability. Kaku (1990) has proposed an empirical formula for damage level prediction based on Van der Meer's experimental data. Smith et al. (1992) have performed hydraulic model tests on the stability of rock slopes by irregular waves. Van der Meer (1992) has carried out an extensive study on stability of rock slopes and gravel beaches and focused on berm breakwaters. Several empirical formulae are developed based on laboratory tests to improve the stability of breakwaters but still there is hardly any development in improving the stability of breakwater.

Mase et al. (1995) and Kim and Park (2005) have used neural network techniques in predicting the stability of rubble mound breakwater. They have used neural network with high epochs upto 50,000. According to them, it is found that neural network predicted better stability values and less armour weights as compared to that of empirical models.

In the present study, back-propagation neural network has been applied to predict the stability of berm breakwater. Four neural network models are constructed based on the input parameters, such as, significant wave height ( $H_s$ ), mean wave period ( $T_m$ ), number of waves ( $N_w$ ), permeability ( $P$ ), damage area ( $S_d$ ), berm width ( $B$ ), armor weight ratio ( $W/W_o$ ). The input parameters of ANN1 model are the parameters of van der Meer's stability formulae. To study the effect of berm width on stability,  $B$  is included in ANN2 model. The effect of armour weight ratio ( $w/w_o$ ) on stability is considered in ANN3 and ANN4 models. The input parameters considered for four network models are shown in Table-1. Estimated stability of breakwater for these network models' result is compared with the theoretical ones. Here the network upto 100 epochs could provide better result than that of Kim and Park (2005) who had used upto 50,000 epochs.

---

<sup>1</sup>Dr, Sr Scientist, National Institute of Oceanography, Goa-403004, India, [smandal@nio.org](mailto:smandal@nio.org),

<sup>2</sup>Dr, Asst Prof, Dept of Applied Mechanics, NITK, Surathkal-575025, India, [sura@nitk.ac.in](mailto:sura@nitk.ac.in);

<sup>3</sup>PG student, Dept of Applied Mechanics, NITK, Surathkal, India, [yr.manjunatha@gmail.com](mailto:yr.manjunatha@gmail.com)

Model	Input Parameters
ANN1	$H_s, T_s, N_w, P, S_d.$
ANN2	$H_s, T_s, N_w, P, S_d, B$
ANN3	$H_s, T_s, N_w, P, S_d, w/w_0.$
ANN4	$H_s, T_s, N_w, P, S_d, B, w/w_0.$

Table 1: ANN input parameters

## 2. STABILITY ESTIMATION

Van der Meer (1988) performed a large number of tests on stability of rubble mound breakwater by changing slope angle, and permeability of breakwater. He also investigated the influence of many parameters on stability like significant wave height, wave period, number of waves, damage level, permeability, relative density of armour unit, etc. as proposed two stability equations which are given below.

For plunging waves,

$$N_s = 6.2 * P^{0.18} * \left[ \frac{S_d}{\sqrt{N_w}} \right]^{0.2} * \frac{1}{\sqrt{\xi_m}} \quad ; \xi_m < \xi_c \quad (1a)$$

For surging waves,

$$N_s = 1.0 * P^{-0.13} * \left[ \frac{S_d}{\sqrt{N_w}} \right]^{0.2} * \cot\theta^{0.5} \xi_m^P \quad ; \xi_m > \xi_c. \quad (1b)$$

Where,  $N_s$  = stability number, the damage level  $S_d$  is defined as the ratio of eroded area of armour units displaced to square of nominal dia of armor ( $D_{n50}$ ),  $\xi_m$  =surf similarity parameter, the transition from plunging to surging waves can be calculated by using a critical value  $\xi_c$

Hudson (1958) defined stability number  $N_s$  as

$$N_s = \left[ \frac{H_s}{\Delta D_{n50}} \right] \quad (2)$$

The experimental data on berm breakwater was generated using a two-dimensional wave flume at National Institute of Technology Karnataka, Surathkal (Rao et al; 2004). A test was carried out for varying berm width with 30% reduced armor weight. The stability of berm breakwater was estimated for 3 wave heights, i.e. 10, 12, 16cms with varying wave periods (1.2, 1.6, 2.0, and 2.6s). These experimental data are used in the present study.

Since there are only three measured values of significant wave height ( $H_s$ ), 3 stability numbers are estimated using Hudson formula [Equation-2] as shown in Fig-1. Whereas stability values predicted by Van der Meer's formulae are not restricted only to three values as the formulae are based on many parameters. The correlation is very less, that is found to be 51.55%. So it is required to improve the estimates of stability values. This can be improved by using artificial neural network.

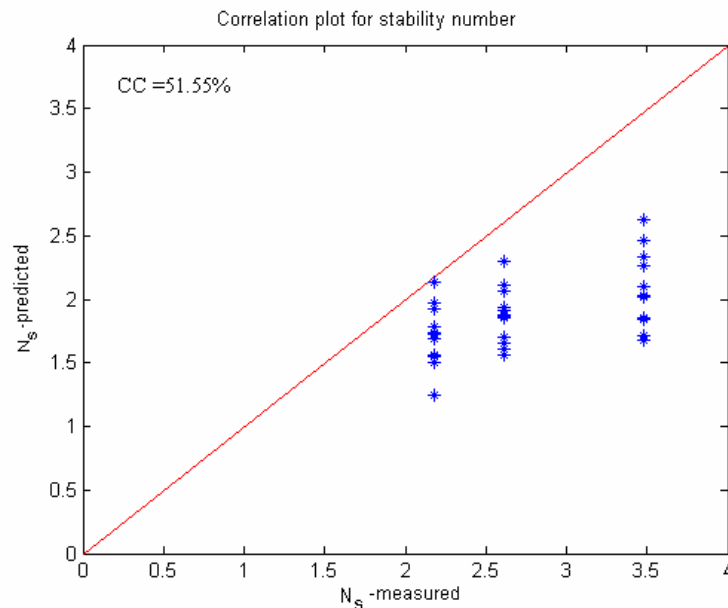


Fig. 1. Stability number predicted (van der Meer) v/s estimated (Hudson).

### 3. NEURAL NETWORK

Artificial neural network (ANN) is an information-processing paradigm that is inspired by the way of biological nervous system, such as brain process information. ANN is composed of large number of highly connected processing element (neurons) working in unison to solve a specific problem. Network learns through examples, so it requires good examples to train properly and further a trained network model can be used for prediction purpose.

In order to allow the network to learn both non-linear and linear relationships between input nodes and output nodes, multiple-layer networks are often used. Among many neural network architectures, the three layers feed forward backpropagation neural network (BNN) is the most commonly used as shown in Fig-2, where  $X_i$ ,  $Y_j$  and  $Z_k$  represent the input, hidden and output nodes respectively.  $W_{ij}$  are the weights between input layer and hidden layer nodes, and  $W_{jk}$  are the weights between hidden layer and output layer nodes.

The back-propagation is a supervised learning technique used for training the neural network. It is most useful for feed forward networks. The back propagation needs to know the correct output for any input parameters. The number of input nodes depends upon the complexity of the problem; in the present study total seven inputs are used to predict the stability of breakwater.

The main objective of BNN technique is to train the model such that the result outputs are nearer to the desired values. Therefore, the error between network output and desired values is minimum.

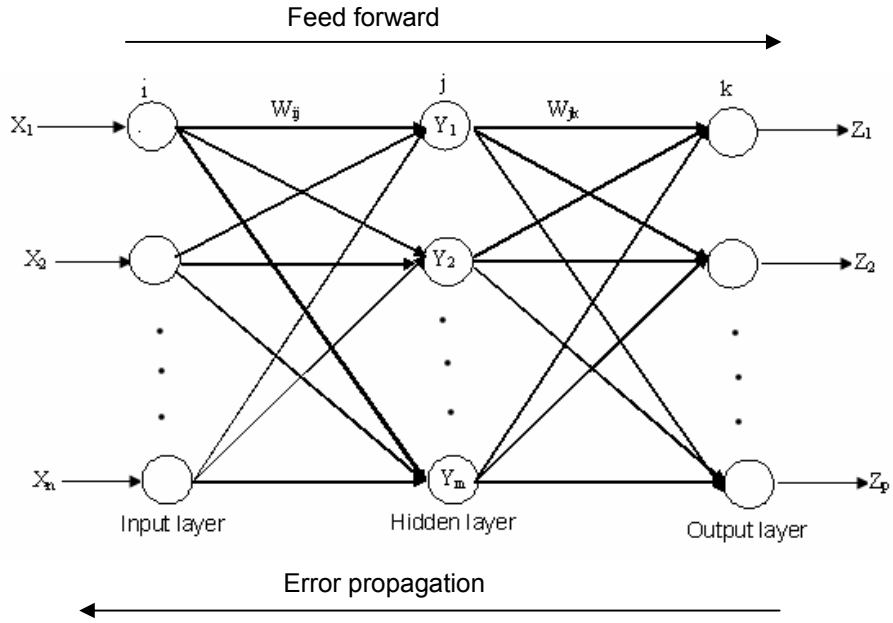


Fig. 2. Three- layers feed forward back propagation neural network.

Mathematically, the feed forward artificial neural network is expressed as

$$Z_k(x) = \sum_{j=1}^m W_{kj} * T_r(y) + b_{ko} \quad (3)$$

$$Y_j = \sum_{i=1}^n W_{ji} * x_i + b_{ji} \quad (4)$$

Where  $x$  is input values from 1 to  $n$ ,  $b_{ji}$  and  $b_{ko}$  are bias values at hidden and output layer respectively.  $m$  is the number of hidden layer nodes and  $T_r(y)$  is transfer function. This transfer function allows a non-linear conversion of summed inputs.

A *non-linear transfer function* is applied between input nodes and hidden nodes. In the present study *Tansig* is used as transfer function, which is expressed as

$$T_r(y) = \left[ \frac{2}{1 + \exp(-2 \times y)} - 1 \right] \quad (5)$$

$y$  is the summation of input values with weights and biases. The transfer function is used to narrow down many input values to single output value. The bias values for both hidden layer and output layer get adjusted for each time of iterations. The weights between hidden and output layers are calculated using updated Levenberg-Marquard algorithm [Wilamowski et al, 2001].

The linear transfer function *purelin* is applied between hidden layer and output layer. And the transfer function expressed as

$$\text{Purelin}(n) = n \quad (6)$$

The overall objective of training algorithm is to reduce the global error, E is defined as,

$$E = \frac{1}{P} \sum_{p=1}^P E_p \quad (7)$$

where p is the total number of training patterns.  $E_p$  is the error at pth training pattern as given by,

$$F(\mathbf{w}) = \sum_{p=1}^P \left[ \sum_{k=1}^K (d_{kp} - o_{kp})^2 \right] \quad (8)$$

Where  $\mathbf{w} = [w_1, w_2 \dots w_N]^T$  consists of all weights of the network,  $d_{kp}$  is the desired value of the  $k^{\text{th}}$  output and the  $P^{\text{th}}$  pattern,  $o_{kp}$  is the actual value of the  $k^{\text{th}}$  output and  $P^{\text{th}}$  pattern, N is the number of weights. Here, Levenberg-Marquardt updated algorithm is used to train the network.

#### 4. STABILITY CALCULATION BY ANN

After training the network model, weights and biases of the network are fixed. These values are shown in Fig-3. The each input value gets multiplied with the weight and adds with bias value. The total sum is the input at each hidden node and pass through a transfer function as defined in Equation 5, and further the output from hidden node get multiplied with the weight and adds with the bias value and total sum pass through *purelin* as shown in Equation 6.

Expressing  $N_i$  as values of hidden nodes and  $F_i$  as transfer function of hidden node i, the stability number is estimated using following formulations:

$$\text{Transfer function } F_i = \left[ \frac{2}{1 + \exp(-2 \times Ni)} - 1 \right] \quad \dots i=1 \text{ to } 4 \quad (9)$$

For ANN2 model, the trained hidden nodes and its transfer functions are

$$N_1 = H_S \times (-14.595) + T_S \times (1.0075) + N_W \times (-5.1509) + P \times (-0.0089) + S_d \times (-0.2753) + B \times (-0.0116) + 1.4494.$$

$$F_1 = (2 / (1 + \exp(-2 \times N_1))) - 1.$$

$$N_2 = H_S \times (-8.4355) + T_S \times (1.2628) + N_W \times (-9.9020) + P \times (-0.0003) + S_d \times (0.0169) + B \times (-0.0002) + 2.2273.$$

$$F_2 = (2 / (1 + \exp(-2 \times N_2))) - 1.$$

$$N_3 = H_S \times (-12.8971) + T_S \times (-0.2440) + N_W \times (-8.0897) + P \times (0.0025) + S_d \times (-0.7564) + B \times (0.0600) + 1.3195.$$

$$F_3 = (2 / (1 + \exp(-2 \times N_3))) - 1.$$

$$N_4 = H_S \times (-1.2112) + T_S \times (0.1906) + N_W \times (-8.0897) + P \times (0.0001) + S_d \times (0.0449) + B \times (-0.0009) + 2.0854.$$

$$F_4 = (2 / (1 + \exp(-2 \times N_4))) - 1.$$

where  $N_1$  to  $N_4$  and  $F_1$  to  $F_4$  represent summation function and transfer function at each hidden node respectively.

The stability number  $N_s$  is computed as:

$$N_s = F_1 \times (0.0961) + F_2 \times (-0.8437) + F_3 \times (0.0403) + F_4 \times (3.9974) - 0.9519 \quad (10)$$

Equation-10 provides trained ANN2 model for estimating stability number ( $N_s$ ) of berm breakwaters.

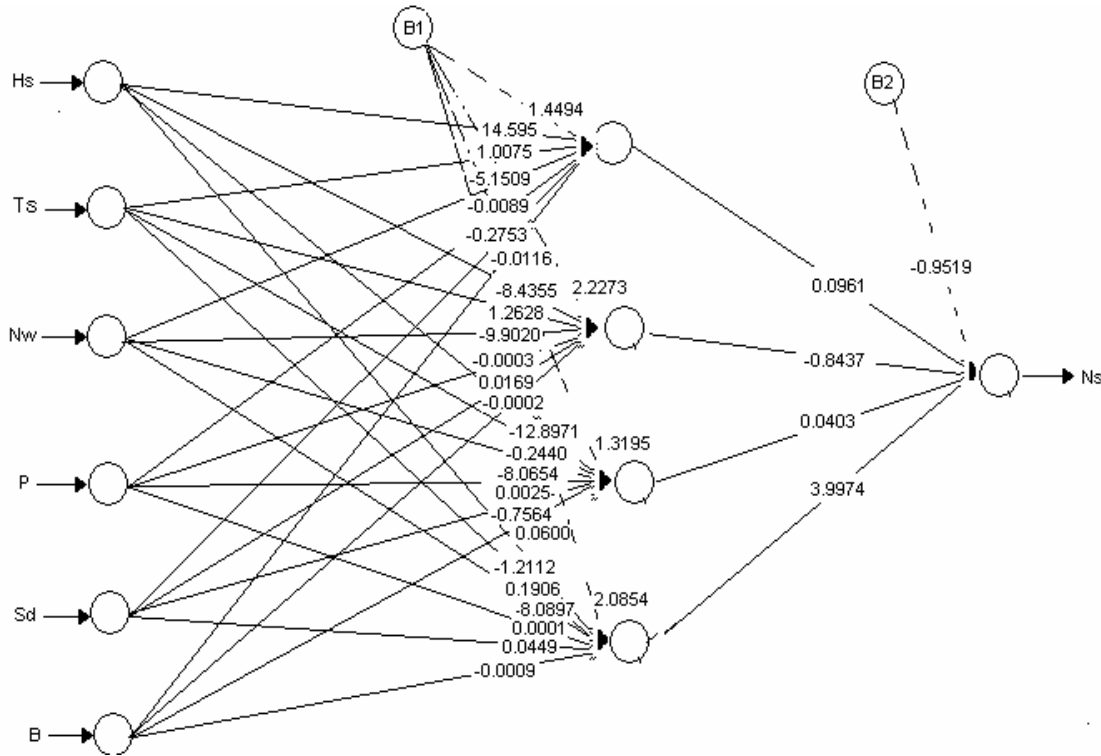


Fig. 3. The ANN2 structure with weights and biases for N-6-4-1.

The correlation coefficient is calculated to know the how best the network predicted stability values are matches with the estimated stability values. The straight line is drawn at an angle of  $45^\circ$  between the two axes to fit the data points. A high correlation is obtained when all the points lies exactly on this straight line.

The network predicted stability number ( $N_s$ ) is calculated using Equation 10. At the end of each training process Correlation Coefficient (CC) is calculated between theoretical (desired) stability number and predicted stability number using following equation

$$CC = \frac{\sum xy}{\sqrt{\sum x^2 \sum y^2}} \quad (15)$$

Where

$x = X - X'$   
 $X$  = predicted stability values by network.  
 $X'$  = mean of  $X$   
 $y = Y - Y'$   
 $Y$  = measured stability values by Hudson and Van der Meer equations.  
 $Y'$  = mean of  $Y$ .

## 5. RESULTS AND DISCUSSION

In neural network technique training of the network plays a very important role, it mainly depends upon updated algorithms to be chosen to train the network. Recently, the study made by Kim and Park (2005) for prediction of stability of rubble mound breakwater using neural network shows some improvement as compared to the study carried out by Mase et al. (1995). Kim and Park (2005) have used steepest gradient approach with epochs upto 50,000.

The parameters that influence the stability of breakwater like significant wave height, wave period, number of waves passing, permeability of breakwater, berm width, damage level and armour weight ratio are considered. Based on these parameters four ANN models were constructed to predict the stability of berm breakwater.

The input parameters of ANN1 model are as same as in van der Meer's stability number for both plunging and surging type of waves. To study the effect of berm width ( $B$ ) on stability,  $B$  is added in ANN2 model and the effect of armour weight ratio i.e.  $w/w_0$  on stability is considered in both ANN3 and ANN4 models. Here, instead of surf similarity parameter used by Kim and Park (2005) to analyze the stability of breakwater, we have used significant height and significant wave period in all four ANN models.

After training and testing of ANN2 network model CCs are calculated between desired output and network output using Equation-10.

In the present study, updated algorithms like Levenberg-Marquardt algorithm (LM), Scaled Conjugate algorithm (SCG) and Gradient Descent algorithm (GDA) [Mller, 1993] are used to train the four models with less number of epochs like 25, 50 and 100. The trained and tested results of the 4 network models using SCG, GDA and LM updated algorithms are shown in Tables 2, 3 & 4. And it is found that the Correlation coefficients (CC) obtained from Levenberg-Marquardt algorithm are close to one and also shows better estimation of stability numbers as compared to that of SCA and GDA methods. The graphs were plotted to show the variation of correlation coefficient with increasing epochs and number of hidden nodes (Fig-4). A high correlation coefficient is mostly obtained at epochs equal to 100 and hidden nodes equal to 6.

Model	Network structure (Input nodes - Hidden nodes - Epochs)	Trained CC	Tested CC
ANN1	5-6-100	0.9789	0.9729
ANN2	6-6-100	0.9082	0.8780
ANN3	6-6-100	0.8082	0.7679
ANN4	7-6-100	0.9269	0.8563

Table 2. Correlation Coefficients for four models using SCG algorithm

Model	Network (Input nodes - Hidden nodes - Epochs)	Trained CC	Tested CC
ANN1	5-6-100	0.8953	0.8265
ANN2	6-6-100	0.8860	0.8072
ANN3	6-6-100	0.8801	0.8724
ANN4	7-6-100	0.8432	0.7061

Table 3. Correlation Coefficients for four models using GDA algorithm

Model	Network (Input nodes - Hidden nodes - Epochs)	Trained CC	Tested CC
ANN1	5-6-100	0.9990	0.9950
ANN2	6-4-50	0.9998	0.9995
ANN3	6-6-100	0.9992	0.9965
ANN4	7-6-100	0.9994	0.9985

Table 4. Correlation Coefficients for four models using LM algorithm

The CCs of all four ANN models with better network structure including Van der Meer model are given in Table-5. The graphs were also plotted between estimated and predicted stability numbers for all network models as shown in Fig-5. The correlation coefficients of trained and tested neural networks are represented as  $CC_{TR}$  and  $CC_{TE}$  respectively. For ANN1 model  $CC_{TR}$  and  $CC_{TE}$  are 0.9603 and 0.9460 respectively. For ANN2 model  $CC_{TR}$  and  $CC_{TE}$  are found to be highest among all four models. Therefore, we can conclude that the berm width is the influencing parameter for the stability of breakwater.

Criteria	VM	ANN1	ANN2	ANN3	ANN4
CCs	0.5155	0.9990	0.9998	0.9992	0.9994
Epochs	-	100	50	100	100
Hidden nodes	-	6	4	6	6

Table-5: Performances of stability for different models



## 6. CONCLUSIONS

Based on the present study the following conclusions were drawn:

The artificial neural network predicted the stability of berm breakwater more accurately compared to the empirical relationships. Since the updated algorithm like Levenberg-Marquardt has shown better improvement in both the training and testing of the network, the results are obtained with very less number of epochs, which makes the process faster and reduces the time required for training the model. Also it is found that the berm width parameter has major influence on the stability of breakwater. Therefore, back-propagation neural network with Levenberg-Marquardt updated algorithm can be effectively used as advanced technique for predicting the stability of berm breakwater.

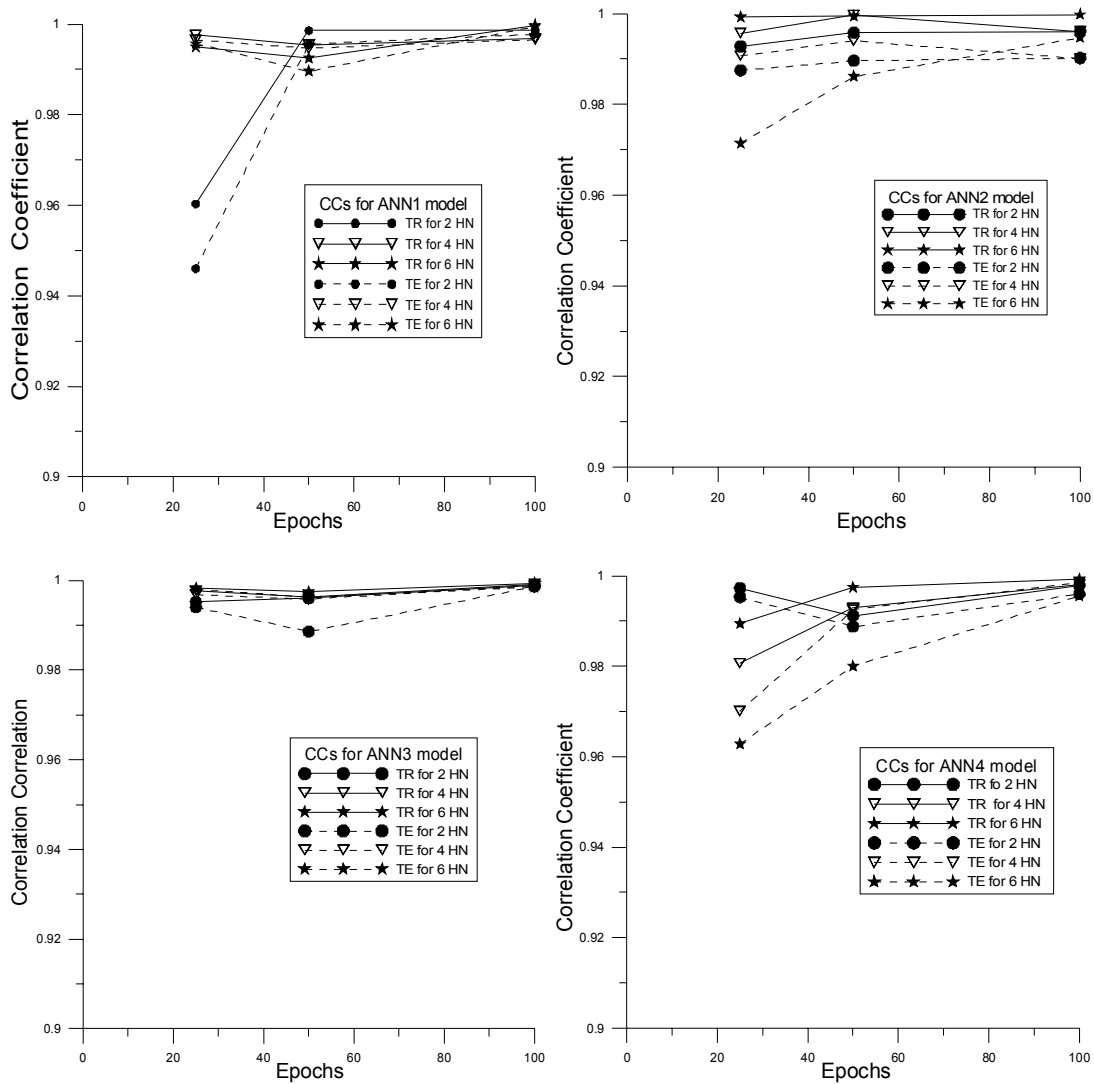


Fig. 4. Variation of Correlation Coefficient v/s Epochs for four ANN models (HN = hidden node, TR = trained, TE= tested)

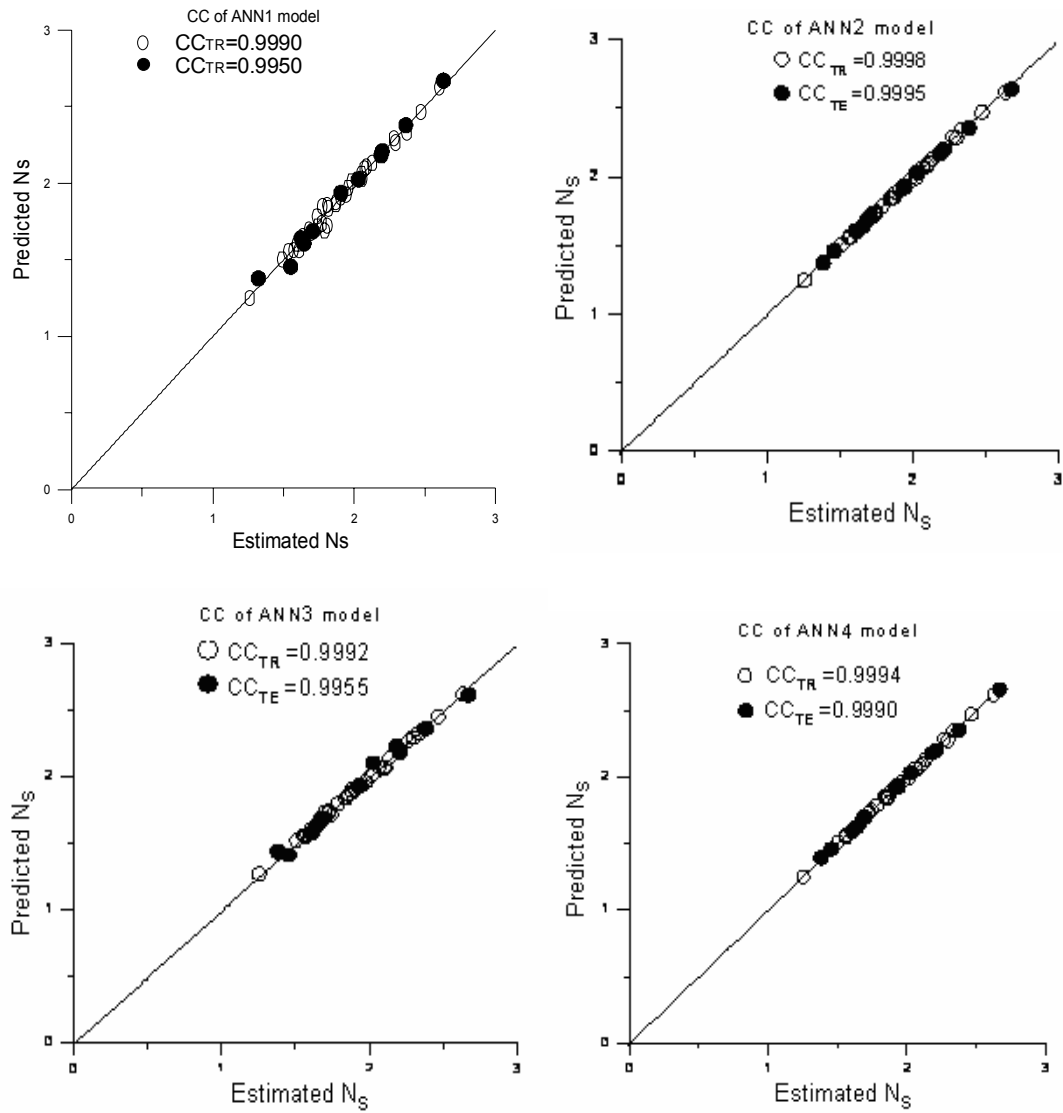


Fig. 5. Variation of estimated  $N_s$  v/s predicted  $N_s$  for four ANN models.

## 7. ACKNOWLEDGEMENTS

The authors are thankful to the Director, NIO, Goa for providing facilities to carry out this work. The authors are grateful to the Head, Dept. of Applied Mechanics and Hydraulics, NITK, Surathkal for providing the experimental data on berm breakwater.

## 8. REFERENCES

- Hudson, R.Y., (1958). Design of quarry stone cover layer for rubble mound breakwaters. Research Report No 2-2. Waterways Experiment Station, Coastal Engineering Research Center, Vicksburg, Miss.
- Hudson, R.Y., (1959). Laboratory investigation of rubble mound breakwaters, Journal of waterways, proc, of ASCE, pp. 93-110.
- Kim, D.H. and Park, W.S., (2005). Neural network for design and reliability analysis of rubble mound breakwater. Ocean Engineering- Elsevier Journal, Vol 32, pp. 1332-1349.
- Kaku S., (1990) Hydraulic stability of rock slopes under irregular wave attack. Master Thesis, University of Delaware, Newark, del.
- Mandal, S.; Rao, S. and Raju, D. H., (2005). Ocean wave parameters estimation using back-propagation neural networks. Marine Structures-Elsevier Journal, Vol 18, pp. 301-308.
- Mase, H.; Masanobu, S. and Tetsuo, S., (1995). Neural network for stability analysis of rubble mound breakwater. Journal of waterway, port, coastal and Ocean Engineering, Vol 121(6), pp. 294-299.
- Mller, M.F., (1993). A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks, Vol-6(4), pp 523-533.
- Rao, S.; Pramod Ch.; BalaKrishna, R., (2004). Stability of berm breakwater with reduced armour stone weight. Ocean Engineering – Elsevier Journal, Vol 31, pp. 1577-1589.
- Rao; S.; Mandal, S. and Prabakaran, N., (2001). Wave forecasting using neural networks. Proc. International Conference ICOE, IIT Madras, pp. 103-108.
- Smith, W. G.; Kuku, S. and Kobayashi, N., (1992) Profile changes of rock slopes by irregular waves. Proc. 23rd International Conference Coastal Engineering ASCE, New York, NY, pp. 1559-1572.
- Van der Meer, J. W., (1988). Rock slopes and gravel beaches under wave attack. PhD Thesis, Delft University of Technology.
- Van der Meer, J. W., (1992). Stability of the seaward slope of berm breakwater. Coastal Engineering-Elsevier Journal, Vol-16, pp. 205-234.
- Wilamowski, B.M.; Iplikci, S; Kaynak, O. And Efe, M.O., (2001). An algorithm for fast convergence in training neural networks. IEEE Transaction on neural networks, pp 1778-1782.