Pergamon

Ocean Engineering 26 (1999) 191–203

OCEAN ENGINEERING

Real time wave forecasting using neural networks

M.C. Deo*, C. Sridhar Naidu

Department of Civil Engineering, Indian Institute of Technology, Bombay, Powai, Mumbai 400 076, India

Received 14 April 1997; received in revised form 28 July 1997; accepted 22 October 1997

Abstract

Forecasting of ocean wave heights, with warning time of a few hours or days, is necessary in planning many operation-related activities in the ocean. Such information is currently derived by numerically solving the differential equation representing wave energy balance. The solution procedure involved is extremely complex and calls for very large amounts of meteorological and oceanographic data. This paper presents a complementary and simple method to make a point forecast of waves in real time sense based on the current observation of waves at a site. It incorporates the technique of neural networks. The network involved is first trained by different algorithms and then used to forecast waves with lead times varying from 3 to 24 h. The results of different training algorithms are compared with each other. The neural output is further compared with the statistical AR models. © 1998 Elsevier Science Ltd. All rights reserved.

Keywords: Wave forecasting; Neural networks; Real time forecasting; Training algorithms; Wave data analysis; Wave heights

1. Introduction

Information on wave heights at a site is required for design as well as operation-related activities in the ocean. While the design exercise calls for wave forecasting over a return period of several years, most of the operation-related works are carried

* Corresponding author.

PII: S0029-8018(97)10025-7

out with the help of wave heights predicted over a couple of hours or days. Operational forecasting of waves is usually made on the basis of past, as well as forecast, data of wind. Conversion of wind information to wave heights is facilitated by various wind–wave relationships. They can be in the form of a parametric equation or a differential equation; the latter being more popular due to its versatility. The differential equation of wave energy balance is required to be solved numerically (WMO, 1988). It is generally employed to produce wave estimates over the next 6–72 h or so. While the spatial wave information provided by numerical wave forecasting schemes is useful and attractive in many applications, it calls for elaborate meteorological and oceanographic data and involves an enormous amount of computational effort. Besides, conversion of wind to waves introduces an element of uncertainty and approximation in the resulting forecasts (Herbich, 1990).

This paper presents a technique to make on-line forecasts of wave heights directly from the observed waves. This is achieved through the use of neural networks. Neural networks have the ability to recognize a hidden pattern in the data and accordingly make a prediction of future values. Provision of model-free solutions, data error tolerance, built-in dynamism in forecasting and lack of any exogenous input requirement make the networks attractive for the present application.

Theoretical concepts related to neural networks can be found in any text book (e.g. Kosko, 1992; Wu, 1994). Recent training algorithms are given in Adeli and Hung (1995). Applications of the networks to prediction problems in civil engineering are relatively recent (Flood and Kartam, 1994a, b). They include forecasting of rainfall (French et al., 1992), runoff (Crespo and Mora, 1993; Karunanithi et al., 1994), concrete strength (Kasperkiewicz et al., 1995), estuarine instabilities (Grubert, 1995) and structural damage (Yeh et al., 1993).

2. The network and training algorithms

2.1. Feed forward network

The network used in the present study is of feed forward type, which has the ability to approximate any continuous function. As shown in Fig. 1, the input nodes receive the data values and pass them on to the first hidden layer nodes. Each hidden node collects the input from all input nodes after multiplying each input value by a weight, attaches a bias to this sum and transforms it through a non-linearity like the sigmoid transfer function. This forms the input to the subsequent hidden layer or to the output layer that operates identically to the first hidden layer. The resulting non-linearly transformed output from each output node constitutes the network output.

2.2. The study area

In the present study, input to the network is in the form of a current wave height value, while the output from it belongs to the forecasted wave height at certain lead time. The site selected is off Yanam, along east coast of India (see Fig. 2). Three-

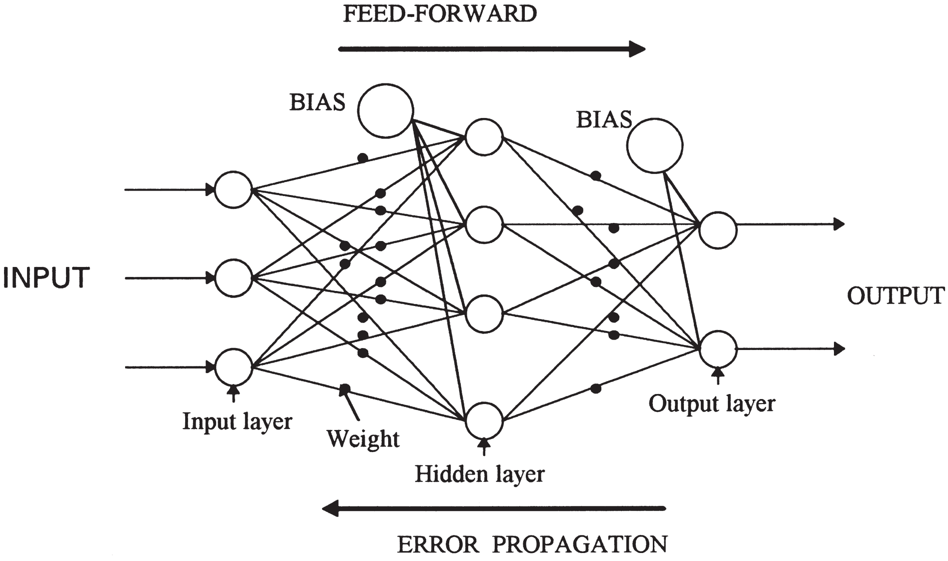

Fig. 1. Feed forward network.

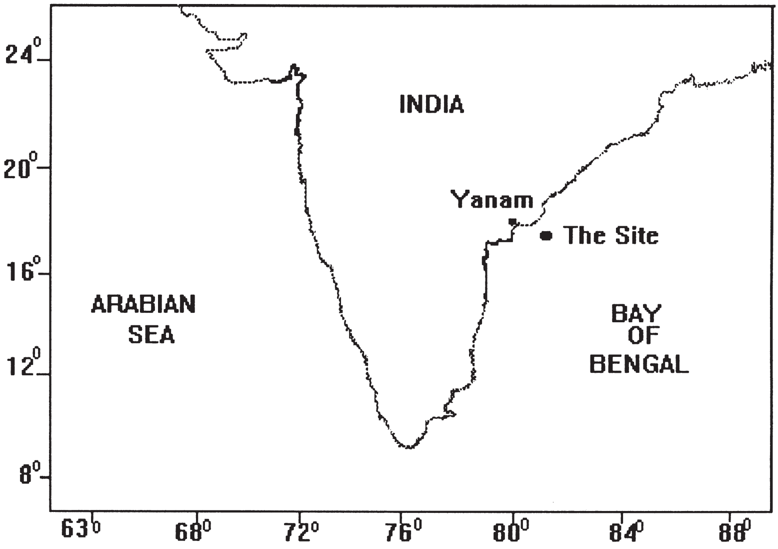

Fig. 2. The site along the Indian coastline.

hourly significant wave heights, *Hs*, deduced from wave rider buoy observations made at this site, were available. The period of data collection was 16 months ranging from May 1983 to August 1984. 80% of the total data was used for network training. The remaining 20% was involved for verification of the training results.

2.3. The training algorithms

The network was trained using three different algorithms. Basically, the objective of training is to reduce the global error, *E*, defined below.

$$E = \frac{1}{P} \sum_{p=1}^{P} E_p \tag{1}$$

where

P = total no. of training patterns
E_p = error for training pattern, p,

$$E_p = \frac{1}{2} \sum_{k=0}^{N} (o_k - t_k)^2 \tag{2}$$

where
N = total no. of training patterns
o_k = network output at the kth output mode
t_k = target output at the kth output mode

In every training algorithm, an attempt is made to reduce this global error by adjusting the weights and biases.

2.3.1. Error back propagation algorithm

This involves minimization of the global error using the steepest descent or gradient descent approach. The network weights and biases are adjusted by moving a small step in the direction of negative gradient of the error function during each iteration. The iterations are repeated until a specified convergence or number of iterations is reached. Mathematically, the gradient descent is given by,

$$\bar{X}_{k+1} = \bar{X}_k - n\bar{g} \tag{3}$$

where

$\bar{X}_{k+1}$ = vector of weight at (k + 1)th iteration index
$\bar{X}_k$ = vector of weights at kth iteration index
n = step size (supplied by the user)
$\bar{g}$ = gradient vector

$$= \nabla f(\bar{X}) \tag{4}$$

where $f(\bar{X})$ = error function, E, for a general weight vector $\bar{X}$

The above error gradient approach is simple to use. However, it converges slowly and may exhibit oscillatory behaviour due to fixed step size. These difficulties can be removed by adopting rather complex algorithm of conjugate gradient. Further, the network architecture of back propagation is required to be prefixed by trial. If the resulting size of the network is too small, it gives rise to under-learning of the problem. On the other hand, lack of generalization and convergence difficulties may arise if the network is too big. The training algorithm of cascade correlation is directed towards removing these problems.

2.3.2. Conjugate gradient algorithm

This technique differs from the previously mentioned error back propagation in gradient calculations and subsequent corrections to weights and bias (Fitch et al., 1991; Fletcher and Reeves, 1964). Here, a search direction, d_k, is computed at each training iteration, k, and the error function, $f(\bar{X})$ is minimized along it using a line search. The gradient descent does not move down the error gradient as in the above back propagation method, but along a direction which is conjugate to the previous step. The change in gradient is taken as orthogonal to the previous step, with the advantage that the function minimization, carried out in each step, is fully preserved due to lack of any interference from subsequent steps. The five step iteration process is as follows (Fitch et al., 1991):

1. Initialize weight vector, $\bar{X}$, by using uniform random numbers from the interval $(\underline{-}\ 0.5, \bar{0}.5)$. Calculate error gradient $\bar{g}_o$ at this point. Select initial search direction $\bar{d}_o = -\bar{g}_o$.
2. For each iteration k, determine constant α_k, which minimizes the error function $f(\bar{X}_k + \alpha_k\bar{d}_k)$ by line search where $\bar{d}_k$ is the search direction at iteration k. Update the weight vector $\bar{X}_k$ *to* $\bar{X}_{k+1}$ using:

$$\bar{X}_{k+1} = \bar{X}_k + \alpha_k d_k \tag{5}$$

3. If error at this iteration, $k + 1$, is acceptable, or if specified number of computations of the function and gradients is reached, terminate the algorithm.
4. Otherwise obtain new direction vector $\bar{d}_{k+1}$. (Fletcher and Reeves, 1964)

$$\bar{d}_{k+1} = -\bar{g}_{k+1} \tag{6}$$

if $k + 1$ is an integral multiple of ‘N’, where N is the dimension of $\bar{X}$. Otherwise,

$$\bar{d}_{k+1} = -\bar{g}_{k+1} + \beta_k\bar{d}_k \tag{7}$$

where

$$\beta_k = \frac{(\bar{g}_k\bar{g}_k^T)}{(\bar{g}_o\bar{g}_o^T)} \tag{8}$$

5. Go to step (ii) for next iteration.

2.3.3. Cascade correlation algorithm

This algorithm starts its training without any hidden node. If the error between the realized output and the targeted one is not low, it adds one hidden node (Fig. 3). This node is connected to all other nodes, except the output nodes. Weights associated with hidden nodes are optimized by a gradient ascent method in which the correlation between the hidden unit's output and the residual error of the network is maximized. If S is an overall sum of such correlations then,

$$S = \sum_{o} \left| \sum_{p} (V_p - \bar{V})(E_{p,o} - \bar{E}_o) \right| \qquad (9)$$

where

o = oth hidden unit
p = pth training pattern
V_p = average of the hidden unit value at pattern p
$\bar{V}$ = average of all V_p values over all patterns
$E_{p,o}$ = residual error for oth hidden unit at pth pattern
$\bar{E}_o$ = average of all $E_{p,o}$ over all patterns

The sum S is maximized by differentiating it with respect to weights and then equating the result to zero.

When a given hidden unit is trained, weights connecting it to the output layer are kept constant. When a new hidden unit is added, its incoming weights are kept unchanged for the remaining training period, during which weights of all links

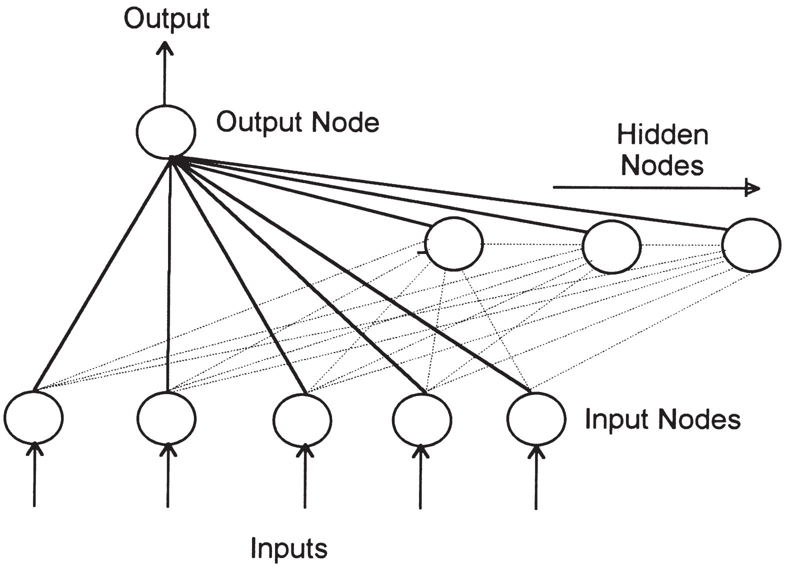

Fig. 3. Cascade correlation learning architecture.

Table 1
Formation of training patterns

No.	Input	Output	No. of training patterns	No. of testing patterns
1	Current three-hourly *Hs*	Next three-hourly *Hs*	2689	671
2	Current three-hourly *Hs*	Next three-hourly *Hs* + next to next three-hourly *Hs*	2689	670
3	Current six-hourly *Hs*	Next six-hourly *Hs*	1345	335
4	Current six-hourly *Hs*	Next six-hourly *Hs* + next to next six-hourly *Hs*	1345	334
5	Current half day average *Hs*	Next half day average *Hs*	670	167
6	Current half day average *Hs*	Next half day average *Hs* + next to next half day average *Hs*	670	166
7	Current day average *Hs*	Next day average *Hs*	337	80

directly connected to output nodes are updated. The addition of hidden unit continues till the desired learning is over (Fahlman and Lebiere, 1990).

3. Results of training and testing

Various training patterns were formed to obtain the desired output in the form of forecasted wave heights at the site off Yanam. Each pattern involved a different combination of input–output values. The forecasts corresponded to different lead times. Different sequences of input–output values were considered. Those combinations that gave better results are presented in Table 1.

This table also gives the total number of training patterns used as well as the total number of patterns utilized for network validation.

Some of the testing results are shown in Figs. 4–7 and in Tables 2–4. The testing

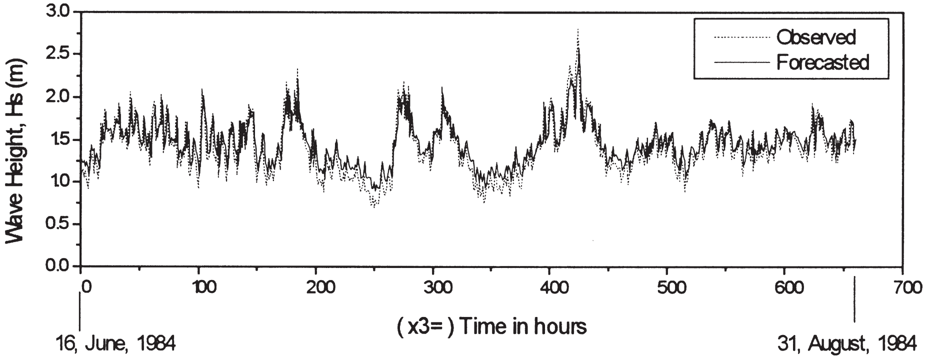

Fig. 4. Three-hourly wave heights. (Algorithm: cascade correlation).

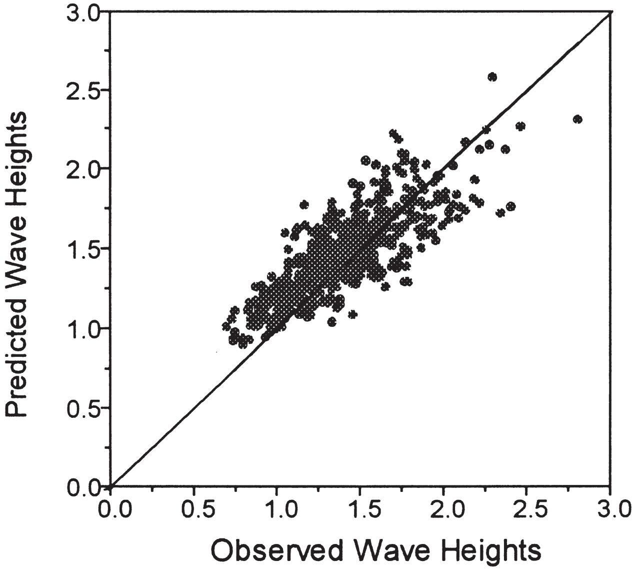

Fig. 5. Three-hour ahead forecasting. (Algorithm: cascade correlation).

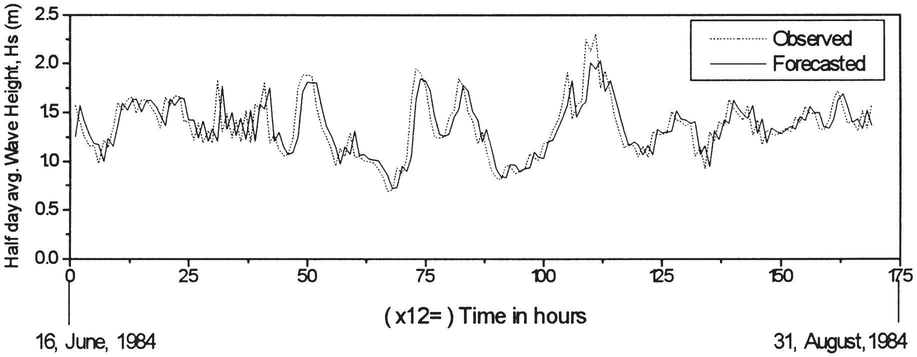

Fig. 6. Half-day average wave heights. (Algorithm: conjugate gradient).

results were studied by (i) comparing the time histories of network forecasts with actual observations, (ii) drawing a scatter diagram of network results vis-à-vis corresponding observations and (iii) computing the correlation coefficient, r, as follows:

$$r = \frac{\Sigma xy}{\sqrt{\Sigma x^2 \Sigma y^2}} \tag{10}$$

where

$x = X - X'$
X = observed wave height

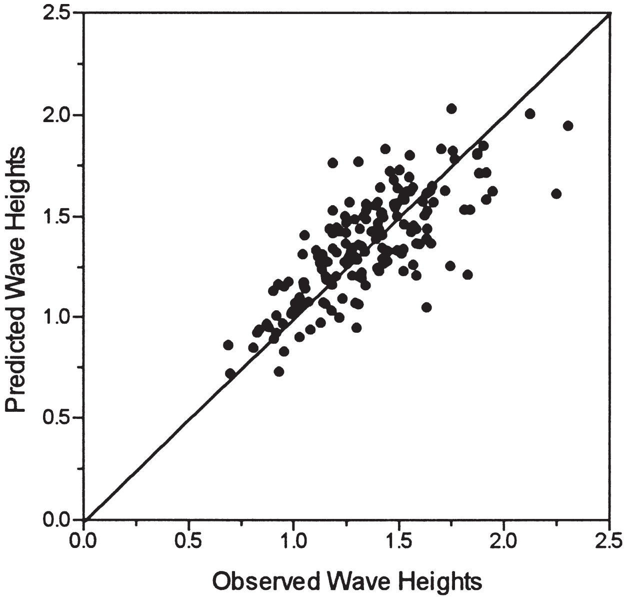

Fig. 7. Half-day average wave heights. (Algorithm: conjugate gradient).

Table 2
Training details for 3 h ahead forecast

Algorithm	Network configuration	Iterations	Time (s)	Time/iteration
Back propagation	1-5-1	67,700	62,050	0.917
Cascade correlation	1-0-1	47	4	0.085
Conjugate gradient	1-2-1	50	1850	37

Table 3
Training details for half-day ahead forecast

Algorithm	Network configuration	Iterations	Time (s)	Time/iteration
Back propagation	1-5-1	11,400	9422	0.826
Cascade correlation	1-0-1	24	4	0.167
Conjugate gradient	1-2-1	50	438	8.76

Table 4
Training details for one-day ahead forecast

Algorith	Network	Iterations	Time (s)	Time/iteration
Back propagation	1-5-1	11,900	7032	0.591
Cascade correlation	1-20-1	2011	22	0.011
Conjugate gradient	1-2-1	50	220	4.4

X' = mean of X
$y = Y - Y'$
Y = forecasted wave height
Y' = mean of $\underline{Y}$

Fig. 4 shows the time history of *Hs* values forecasted for the next 3 h on the basis of current three-hourly *Hs* observations. Underlying training algorithm is cascade correlation. The corresponding scatter diagram is given in Fig. 5. It may be noted that the rising and falling tendencies of observed wave heights are fairly picked up in the neural output and that the observations and predictions are fairly close to each other. This is further confirmed by the corresponding high value of correlation coefficient, *r*, which was 0.81. Considering the randomness in the occurrence of waves, great accuracy in forecasting is very difficult to achieve. Validation of the numerical wave forecasting models also generally yield similar results (WMO, 1988).

Forecasting of next half-day wave height on the basis of its current value can be seen in Figs. 6 and 7. The underlying algorithm is conjugate gradient. It appears from these figures that as the lead time of forecast increases, forecasting accuracy reduces, which is further reflected in the value of the correlation coefficient that reduces from 0.81 for next three-hourly forecasts to 0.78 for next half-day forecasts and further to 0.71 for next daily mean forecasts.

The configuration of the network used, total number of iterations required to achieve the specified minimum global error of 0.01%, as well as the time required for training in above studies, are given in Tables 2–4. The cpu time indicated there pertains to a PC-AT 486 machine. Table 2 indicates training details for 3 h ahead wave forecasting, while the details for half-day and daily wave forecasting are shown in Tables 2 and 3. The first and last values separated by dashes under the column ‘Network configuration’ indicate number of input and output nodes involved, while the middle value gives the number of hidden nodes.

It may be seen from these tables that the algorithm of cascade correlation takes the least time for training, which is only a very small fraction of the time taken by the other two training schemes. The flexibility in selecting the network architecture thus seems to be paying rich dividends in this application. Even though the back propagation method involves a large number of training iterations, it is the conjugate gradient technique that takes the longest time to complete one training cycle. These observations are important for forecasting in dynamic or adaptive mode. The coefficient of correlation did not vary significantly across the training algorithms.

3.1. Usefulness and limitations

It is emphasized that the real time wave forecasting discussed above is not a substitute to the operational numerical models used world over. Unlike these techniques, the proposed method is site specific and is applicable only when on-line wave data is being gathered at the site, say, through a pressure gauge fitted on an oil platform. In such a case, decision regarding any wave-sensitive repair work on the platform, lasting for next few hours or so, can be taken based upon the method presented. For such specific forecasting requirements, use of the complex numerical wave model, calling for large amount and types of exogenous data and yielding unwanted spatial information, may not be necessary. The proposed technique is computationally much simpler and can be made dynamic and adaptive as and when new data are incorporated in training.

3.2. Use of AR models

The technique of neural networks discussed in the previous sections involves use of the observations at current time step to forecast their future time step values. Statistical methods like Auto Regressive (AR) also work similarly, but they are model-based unlike neural networks. The neural output was therefore compared with the AR model output in the present studies. The AR model of order p, i.e. AR(p), is given by:

$$a_i = \phi_{p,1}a_{i-1} + \phi_{p,2}a_{i-2} + \phi_{p,3}a_{i-3} + \dots\dots\dots + \phi_{p,p}a_{i-p} + n_i \tag{11}$$

where a_i = ith variable of the sequence of zero mean and unit variance

n_i = random number at ith time
$\phi_{p,1},\phi_{p,2},\dots$ = auto regressive parameters

The variable a_i is related to the wave height x_i at ith time by:

$$x_i = \bar{X} + S\,a_i \tag{12}$$

where $\bar{X}$ and S are the mean and variance of the observed or historical wave height values.

The auto regressive parameters are determined with the help of auto correlation coefficients of the observed series (Box and Jenkins, 1970).

The forecasts of significant wave heights corresponding to varying warning times were obtained from AR models of different orders. Their comparison with corresponding observations indicated that the AR models have slightly lower accuracy than the neural networks. They also involve larger lags with respect to the observations. Fig. 8 shows a typical AR(2)-based forecasting of half-day average wave heights vis-à-vis corresponding observations. The AR models exhibited slightly lower values of the correlation coefficient than the neural network. These values were 0.78, 0.72 and 0.70 for 3 h, 12 h and 24 h ahead forecasts, respectively (for AR(2) model fits), while the corresponding figures were 0.81, 0.78 and 0.71 for the

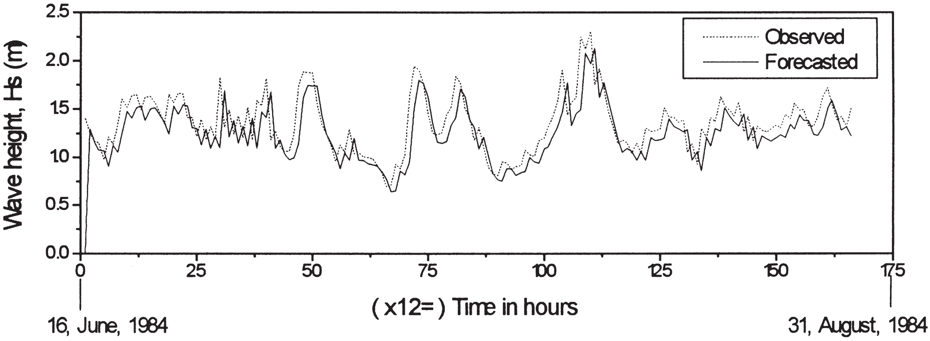

Fig. 8. Half-day average wave heights. (Model: AR-2).

neural network forecasts. Higher order models than AR(2) were not found to improve the results.

It may be noted that the neural network has many practical advantages compared with statistical models, like the AR. Basically, it is model-free, unlike the AR techniques, and hence is more general, flexible and adaptable. It has the ability to produce simultaneous forecasts for multiple time steps in future with much ease compared with the AR models.

4. Conclusions

The foregoing sections presented a technique to make operational wave forecasts using neural networks. It carries with it all advantages of neural computations over conventional deterministic or statistical models. The technique was employed to forecast significant wave heights with warning times of 3–24 h at an offshore site where wave height measurements were available.

Lower warning time was found to produce better forecasts of significant waves.

A comparison of different training algorithms indicated superiority of the cascade correlation scheme in terms of faster convergence. The neural networks exhibited higher covariation of the wave height predictions with their actual observations than the AR models.

Acknowledgements

The authors thank Mr Konda Thirumalaiah, research scholar, for computational help and Mr Matt White of CMU for providing public domain access to Cascade correlation software.

References

Adeli, H., Hung, 1995. Machine Learning, Neural Networks, Genetic Algorithms and Fuzzy Systems. Wiley, New York.

Box, G.E.P., Jenkins, G.M., 1970, Time Series Analysis, Forecasting and Control. Holden Day, San Francisco.

Crespo, J.L., Mora, E., 1993. Drought estimation with neural networks. Advances in Engineering Software 18, 167–170.

Fahlman, S.E., Lebiere, C., 1990. The cascade corrlation training architecture. In Advances in Neural Engineering Processing Systems, vol. 2. Morgan Kaufmann, San Mateo, CA.

Fitch, J.P., Lehman, S.K., Dowla, F.U., Lu, S.K., Johansson, E.M., Goodman, D.M., 1991. Ship wake detection procedure using conjugate gradient trained artificial neural network. IEEE Transactions on Geosciences and Remote Sensing 9 (5), 718–725.

Fletcher, R., Reeves, C.M., 1964. Function minimization by conjugate gradients. Computer Journal 149–153.

Flood, I., Kartam, N., 1994a. Neural networks in civil engineering 1: principles and understanding. ASCE Journal of Computing in Civil Engineering 8 (2), 131–148.

Flood, I., Kartam, N., 1994b. Neural networks in civil engineering 2: systems and application. ASCE Journal of Computing in Civil Engineering 8 (2), 149–162.

French, M.N., Krajewski, W.F., Cuykendall, R.R., 1992. Rainfall forecasting in space and time using a neural network. Journal of Hydrology 137, 1–29.

Grubert, J.P., 1995. Prediction of estuarine instabilities with artificial neural networks. ASCE Journal of Computing in Civil Engineering 9 (4), 266–274.

Herbich, J.B., 1990. Handbook of Coastal and Ocean Engineering, vol. 1. Gulf, London.

Karunanithi, N., Grenney, W.J., Whitley, D., Bovee, K., 1994. Neural networks for river flow prediction. ASCE Journal of Computing in Civil Engineering 8 (2), 279–284.

Kasperkiewicz, J., Racz, J., Dubrawski, A., 1995. HPC strength prediction using neural network. ASCE Journal of Computing in Civil Engineering 9 (4), 279–284.

Kosko, B., 1992. Neural Networks and Fuzzy Systems. Prentice Hall, New Jersey.

Yeh, Y.C., Kuo, Y.H., Hsu, D.S., 1993. Building KBSE for diagnosing PC piles with artificial neural networks. ASCE Journal of Computing in Civil Engineering 7 (1), 71–93.

WMO, 1988. Guide to Wave Analysis and Forecasting. No. 702, Secretariat of the World Meteorological Organization, Geneva.

Wu, J.K., 1994. Neural Networks and Simulation Methods. Marcel Dekker, New York.