Longshore current velocities prediction: using a neural networks approach

T. M. Alaboud & M. S. El-Bisy
Civil Engineering Dept., College of Engineering and Islamic Architecture, Umm Al_Qura University, Makkah, K.S.A.

Abstract

Accurate current conditions prediction and supplement is an important task in the successful development and management of coastal zone infrastructure. Longshore current velocities forecasting is currently made by adopting conventional numerical and analytical models. In fact, longshore current velocities prediction in the conventional numerical and analytical models require a large amount information apart from historical wave observations and topography maps and are complex and tedious to apply specifically when point-forecasts at specific locations are needed. Therefore, this paper presents an application of a neural network for forecasting and supplementing the daily and monthly longshore current velocities. The neural network was trained using back propagation and cascade correlation algorithms. The data of five stations along the Damietta promontory on the Egyptian Nile delta coast were used to test the performance of the neural network model. The results indicated that the neural network can efficiently forecast longshore current velocities. Neural network forecasting was also found to be more accurate than traditional statistical time-series analysis.

Keywords: longshore current velocities, forecasting, neural networks, ARIMA model.

1 Introduction

Longshore currents play the main role in sediment transport along the coast. The longshore current combined with the agitating action of the breaking waves, are the primary factors in causing sand movement. Studies indicate that the greatest

doi:10.2495/CP110161

percentage of sand transport along a coast occurs shoreward of the breaker zone [1].

The strength of the littoral current in terms of the wave and beach characteristics for a relatively straight shoreline has been studied by several investigators, [2–6], and the subject is so important that many analytical solutions of the governing equations are published almost yearly. As analytical solutions are made more general, however, their evaluation becomes mathematically and computationally complex.

The alternative approach to analytical models of the longshore current is to solve the governing equations numerically. The numerical models are more computationally efficient than the complex analytical solutions. While simulations space wise information yielded by these models is advantageous, they require excess information apart from historical wave observations and topography maps and are complex and tedious to apply specifically when point-forecasts at specific locations are needed [1].

In a situation when a considerable amount of data about the studied processes is available, the use of traditional techniques may hardly be justified. In order to achieve optimal result it is important to use all available data. One of the options is to use regression – and auto-correlation – based statistical method like ARIMA models. However, with the recent advancements in artificial intelligence, data mining and soft computing, there is a choice of better techniques that may help to solve the problem. One of them is neural networks (NNs) which may be considered as a class of models which form a numerical modeling point view, are a general framework for representing non-linear mappings between multi-dimensional spaces in which the form of the mapping is governed by a number of adjustable parameters. It’s by the modification of the adjustable parameters that the NNs model “learns” or identifies the mapping [7].

The present study aims to producing a reliable neural methodology applicable to the tasks of site-specific longshore current velocities estimations. The data base of Damietta promontory in Egypt was used to test the performance of the different models.

2 Time series forecasting models

2.1 ARIMA model

Introduced by Box and Jenkins [8], the ARIMA model has been one of the most popular approaches to forecasting. In an ARIMA model, the future value of a variable is supposed to be a linear combination of past values and past errors, expressed as follows:

$$\begin{aligned} y_t &= \theta_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots. + \phi_p y_{t-p} \\ &\quad + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots. - \theta_q \varepsilon_{t-q} \end{aligned} \tag{1}$$

where y_t is the actual value and ε_t is the random error at time t, ϕ_i and θ_j are the coefficients, p and q are integers that are often referred to as autoregressive and moving average polynomials, respectively. Basically, this method has three phases: model identification, parameter estimation and diagnostic checking.

In the identification phase, data transformation is often needed to make the time series stationary. Stationary is a necessary condition in building an ARIMA model that is useful for forecasting. A stationary time series has the property that its statistical characteristics such as the mean and the autocorrelation structure are constant over time. When the observed time series presents trend and heteroscedasticity, differencing and power transformation are often applied to the data to remove the trend and stabilize the variance before an ARIMA model can be fitted. Once a tentative model is specified, estimation of the model parameters is straightforward. The parameters are estimated such that an overall measure of errors is minimized. This can be done with a nonlinear optimization procedure.

The last phase of model building is the diagnostic checking of model adequacy. This is basically to check if the model assumptions about the errors, ε_t, are satisfied. Several diagnostic statistics and plots of the residuals can be used to examine the goodness of fit of the tentatively entertained model to the historical data. If the model is not adequate, a new tentative model should be identified, which is again followed by the steps of parameter estimation and model verification. Diagnostic information may help suggest alternative model(s).

This three phase's model building process is typically repeated several times until a satisfactory model is finally selected. The final selected model can then be used for prediction purpose.

2.2 Neural networks model

NNs are flexible computing frameworks for modeling a broad range of nonlinear problems. One significant advantage of the NN models over other classes of nonlinear model is that NNs are universal approximators which can approximate a large class of functions with a high degree of accuracy. Their power comes from the parallel processing of the information from the data. No prior assumption of the model form is required in the model building process. Instead, the network model is largely determined by the characteristics of the data.

Single hidden layer feedforward (FF) network is the most widely used model form for time series modeling and forecasting [9]. The model is characterized by a network of three layers of simple processing units connected by acyclic links. The relationship between the output (y_t) and the inputs (y_{t-1}, $y_{t-2,\dots\dots,}$ y_{t-m}) has the following mathematical representation:

$$y_t = \alpha_0 + \sum_{j=1}^{n} \alpha_j f\left(\sum_{i=1}^{m} \beta_{ij} y_{t-i} + \beta_{0j} \right) + \varepsilon_t \qquad (2)$$

where m is the number of input nodes, n is the number of hidden nodes, $\alpha_j(j$=0,1,...,n) and B_{ij} (i=0,1,...,m; j= 0,1,..., n) are the model parameters often called the connection weights. α_0 and B_{oj} are weights of arcs leading from the bias terms. The logistic function is often used as the hidden layer transfer function, that is,

$$g(x) = \frac{1}{1 + \exp(-x)} \tag{3}$$

Hence, the ANN model of eqn. (2) in fact performs a nonlinear functional mapping from the past observations ($y_{t-1}, y_{t-2,\dots\dots,} y_{t-m}$) to the future value y, i.e.,

$$y_t = f\left(y_{t-1}, y_{t-2}, \dots, y_{\mathrm{t-p}}, w\right) + \varepsilon_t \tag{4}$$

where w is a vector of all parameters and f is a function determined by the network structure and connection weights. Thus, the neural network is equivalent to a nonlinear autoregressive model. Note that eqn. (2) implies one output node in the output layer which is typically used for one-step-ahead forecasting.

Many alternative training processes are available, out of which the present study adopted two schemes, back propagation (BP) and cascade correlation (CC).

2.2.1 Back propagation algorithm

The goal of any training algorithm is to minimize the global (mean sum squared) error E, defined below:

$$E = \frac{1}{2}\sum \left(O_n - P_n\right)^2 \tag{5}$$

where O_n is the observations, and P_n is the predictions for any n output node. The summation has to be carried out over all output nodes for every training pattern. A pair of input and output values constitutes a training pattern.

The BP algorithm calculates the error using eqn. (5), and then it adjusts the weights, first in the output layer, and then distributes it backward from the output to hidden and input nodes. This is done using the steepest gradient descent principle where the change in weight is directed towards negative of the error gradient, i.e.

$$\Delta w_n = \alpha \Delta w_{n-1} - \eta \frac{\partial E}{\partial w} \tag{6}$$

where w is the weight between any two nodes; Δw_n, Δw_{n-1} are the changes in this weight at n and n-1 iteration, α the momentum factor, and η is the learning rate.

2.2.2 Cascade correlation algorithm

The cascade correlation (CC) is another network that modifies its own architecture as training progresses. CC network starts with a minimal topology, consisting only of the required input and output units. This net is trained until no further improvement is obtained.

Next, one hidden node is added to the net in a two-step process. During the first step, a candidate unit is connected to each of the input units, but is not connected to the output units. The weights on the connections from the input units to the candidate unit are adjusted to maximize the correlation between the candidate's output and the residual error at the output units. The residual error is the difference between the target and the computed output, multiplied by the derivative of the output unit's activation function, i.e., the quantity that would be propagated back from the output units in the back propagation algorithm. When this training is completed, the weights are frozen and the candidate unit becomes a hidden unit in the net.

The second step in which the new unit is added to the net now begins. The new hidden unit is then connected to the output units, and the weights on the connections being adjustable. Now all connections to the output units are trained. Training of this node is based on maximization of overall correlation *S* given by:

$$S=\sum_{o}\left[\sum_{p}\left(V_p-\bar{V}\right)\left(E_{p,o}-\bar{E}_o\right)\right] \tag{7}$$

where V_p is the output of the new hidden node for pattern p, $\bar{V}$ is the average of V_p over all patterns, $E_{p,o}$ is the network output error for output node o on pattern p training pattern, and $\bar{E}_o$ is the average of network error over all patterns.

The second step in which the new unit is added to the net now begins. The new hidden unit is then connected to the output units, and the weights on the connections being adjustable. Now all connections to the output units are trained. The addition of hidden nodes continuous until the desired learning is over.

Details of the concept of neural networks and the description of various training algorithms can be seen in References [10–12].

2.3 Performance evaluation

To examine how close of the predict to the real value of longshore current velocities, four indices, mean absolute percentage error *MAPE*, root mean square error *RMSE*, scatter index *SI*, and correlation coefficient *R*, were employed to evaluate the performance of neural networks and ARIMA model:

$$MAPE=\left[\frac{1}{N}\sum_{i=1}^{N}\left|\frac{P_i-O_i}{O_i}\right|\right]\times 100 \tag{8}$$

$$RMSE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(P_i-O_i\right)^2} \tag{9}$$

$$SI=\frac{RMSE}{\bar{O}}\times 100 \tag{10}$$

$$R=\frac{\sum_{i=1}^{N}\left(P_i-\bar{P}\right)\left(O_i-\bar{O}\right)}{\sqrt{\sum_{i=1}^{N}\left(P_i-\bar{P}\right)^2\sum_{i=1}^{N}\left(O_i-\bar{O}\right)^2}} \tag{11}$$

where O_i is the observed value, P_i is the predicted value, N is the total number of data points in validation, $\bar{O}$ is the mean value of observations, and $\bar{P}$ is the mean value of predictions.

3 Data acquisition

There are twelve stations along the two promontories of the Egyptian Nile delta coast for measuring the longshore currents. Seven stations lie along both sides of Rosetta promontory and five stations along Damietta promontory. The longshore current measurements are taken two times per day. The longshore currents data was recorded in the surf zone, which is bounded by the shore and the breaker lines at depths ranging between 1.2 m to 1.5m. The measurements were taken by Coastal Research Institute, CRI over a period of 16 years ranging from 1982 to 1997. The observations from five stations along Damietta promontory were used in this study.

3.1 Pre-processing of data

NNs have been shown to be able to process data from a wide variety sources. They are however, only able to process the data in a certain format. Furthermore the way the data is presented to the network affects the learning of the network. Therefore, a certain amount of data processing is required before presenting the training patterns to the network. In this study a linear scaling was used.

One of the reasons for pre-processing the output data is that a sigmoidal transfer function is usually used within the network. Upper and lower limits of output from a sigmoid transfer function are generally 1 and 0 respectively. Scaling of the inputs to the range $[-1,+1]$ greatly improves the learning speed, as these values fall in the region of sigmoid transfer function where the output is most sensitive to variations of the input values. It is therefore recommended to normalize the input and output data before presenting them to the network. Scaling data can be linear or non-linear, depending on the distribution of the data. Most common functions are linear and logarithmic functions. A linear normalization function within the values of zero to one is:

$$S = (V - V_{\min})/(V_{\max} - V_{\min}) \tag{12}$$

where S is the normalised value of variable V, V_{min} and V_{max} are variable minimum and maximum values respectively.

4 Results and discussion

4.1 Monthly forecasts

4.1.1 NNs model

The determination of the optimal network architecture for a given task remains an open research question. Network size is determined by input number m, the number of hidden layers and the number of neurons in the hidden layers. If different network sizes have similar values for the error function, the smallest network size is optimum.

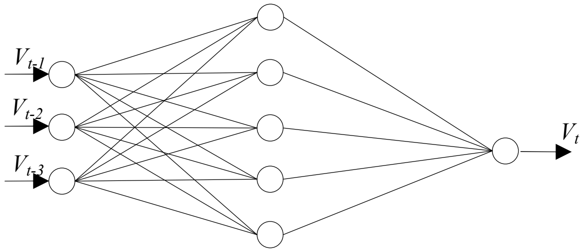
Many techniques are proposed to determine optimum neural networks such as the ad hoc approach, the dynamic approach, and the distribution approach. In this study trails were used to characterize the deterministic and random properties of the data to determine optimum neural networks.

The longshore current data of Damietta promontory pertaining to the first thirteen years (from 1982 to 1994) was used for training while the observations of the last three years (from 1995 to 1997) were reversed for testing the network. It was found that the minimum input number of dynamic variables needed to train the networks equal three preceding values. At the length of input sequence (*m*) equal 3, there is a maximum statistical independence between distance observations of data series, so, $m = 3$ maximize the network ability to learn from the data and should be the optimum number of nodes in the input layer. To verify the optimum input number, *RMSE* corresponding to various input number of nodes is calculated. It is found that neural networks corresponding to $m = 3$ give minimum *RMSE*.

To finally determine the optimum size of neural networks based on the determined input numbers, the networks with one hidden layer were used for training and test by changing neuron size in the hidden layer. It is found that the network with one layer and five neurons can make RMSE converge to lower values. In the case of the alternative CC scheme, hidden layers and neurons get automatically fixed during training as explained in the sec. 2.2.2.The neural network models were trained using backpropagation and cascade correlation and all computations were performed within the MATLAB version 6.1.0.450 release 12.1 (The Math Works Inc. May18, 2001). Figure 1 represents the topology of the optimum network trained using BP algorithm. In this figure $y_t, y_{t-1, \ldots,} y_{t-3}$denote longshore current velocity at time *t*, *t*-1, ..., *t*-3, respectively.

For the developed BP network mentioned earlier, the results tested instances for monthly longshore current velocities were shown in Figure 2. The predictions can be seen as fairly close to the corresponding actual measurements. All ups and downs in the observed time series may well be modeled well in the predicted series. It can be observed that a maximum absolute percentage error of 9.18%, a minimum absolute percentage error of 0.17% and the MAPE of 4.87% were obtained between the predicted and the observed series. The correlation coefficient of 0.957 was obtained for longshore current velocities prediction.

The longshore current velocities forecasting as above based on FF-BP was repeated by using the CC scheme of training. . This was done to see whether the training improves by adopting a different algorithm. It was found that a maximum absolute percentage error of 8.34%, a minimum absolute percentage error of 0.18% and the MAPE of 4.43% were obtained between the predicted and the observed series. The correlation coefficient of 0.965 was obtained for longshore current prediction. Results of the testing of this network indicated almost same performance of the PB training method.



Input Layer Hidden Layer Output Layer

Figure 1: Structure of the artificial neural network.

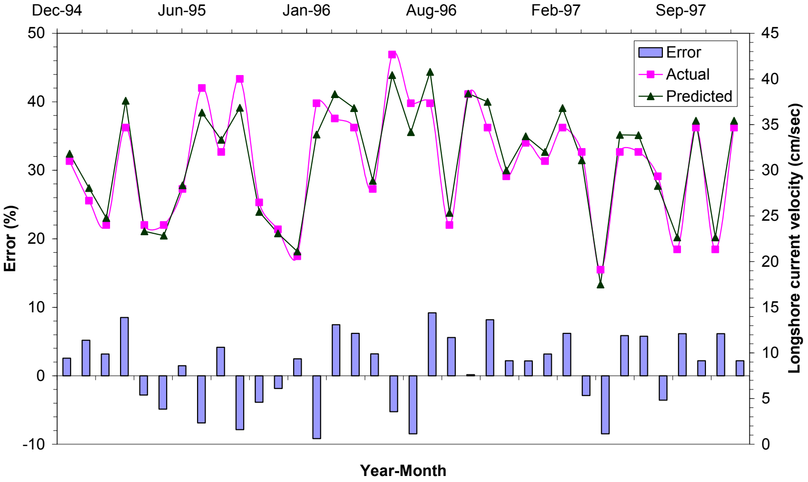


Figure 2: Time series plot of monthly longshore current velocities observed and predicted by FF-PB.

4.1.2 ARIMA model

The ARIMA model was implemented via SAS/ETS software version 6, a component of the SAS system (SAS Institute, Inc., 1992). This model was used to forecast longshore current velocities. Similar to the case of neural networks the model was calibrated using the first 13 years data and tested on the last 3 years observations. Considerable experimentation was made to arrive at the most satisfactory calibration. Figure 3 represent the time series plot of monthly longshore current velocities observed and predicted. It can be observed that a maximum absolute percentage error of 14.69%, a minimum absolute percentage error of 1.23% and the MAPE of 6.801% were obtained between the predicted and the observed series. The correlation coefficient of 0.931 was obtained for longshore current velocities prediction.

Table 1 compares the forecasting results of different models. Those results indicate that the neural networks model (PB and CC) outperforms the ARIMA model in terms of four indices, revealing that the ARIMA model cannot capture all of the patterns in the longshore current velocities. The neural networks model is, however, can significantly reduce the overall forecasting errors. Furthermore, the results indicate the proposed CC model is superior to PB model. The development of the longshore current velocities scheme in a model free manner seems to have resulted in more flexibility in the data mining approach of neural network than one associated with fixed type of stochastic time series schemes (ARIMA). In terms of RMSE, the percentage improvements of the CC scheme over ARIMA and FF-BB for monthly forecasting were 51.88% and 10%, respectively.

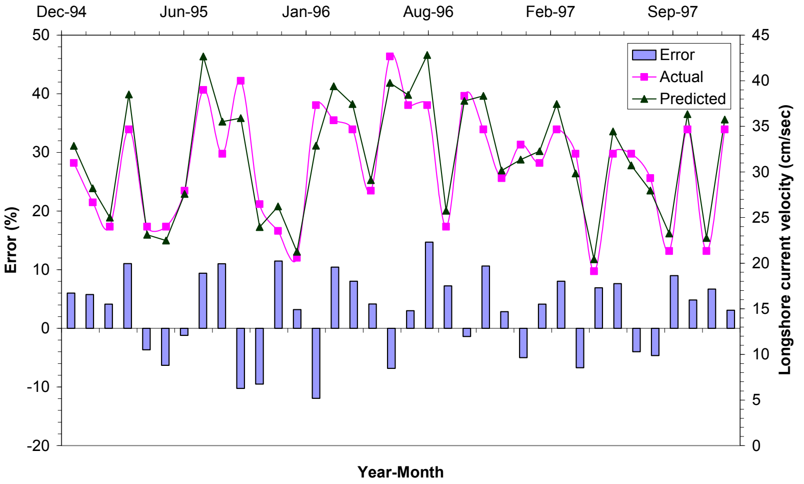


Figure 3: Time series plot of monthly longshore current velocities observed and predicted by ARIMA.

Table 1: Evaluating performance of models.

Forecast	Scheme	MAPE (%)	RMSE	SI (%)	R
Month	FF-BP	4.873	5.461	17.739	0.957
	FF-CC	4.43	4.964	16.126	0.965
	ARIMA	6.801	7.535	24.477	0.931
Daily	FF-BP	8.68	9.404	32.816	0.951
	FF-CC	7.548	8.178	28.536	0.962
	ARIMA	11.677	12.943	45.163	0.927

4.2 Daily forecasts

Here also, like in the case of monthly longshore current velocities forecasting FF network trained using BP and CC algorithms were developed using the first thirteen years data and tested using three years data (as well as ARIMA schemes for compression purpose). Figure 4 shows the performance of neural networks with respect to testing a set of longshore current data belonging to the last 3 years of measurement, while Figure 5 shows the same for the ARIMA model. Here also, like the previously described monthly predictions, the neural network

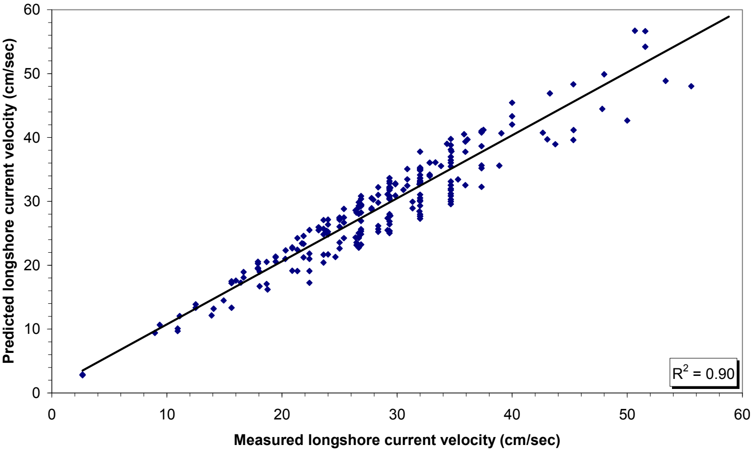


Figure 4: Scatter plot comparing observed and predicted daily longshore current velocity by FF-BP.

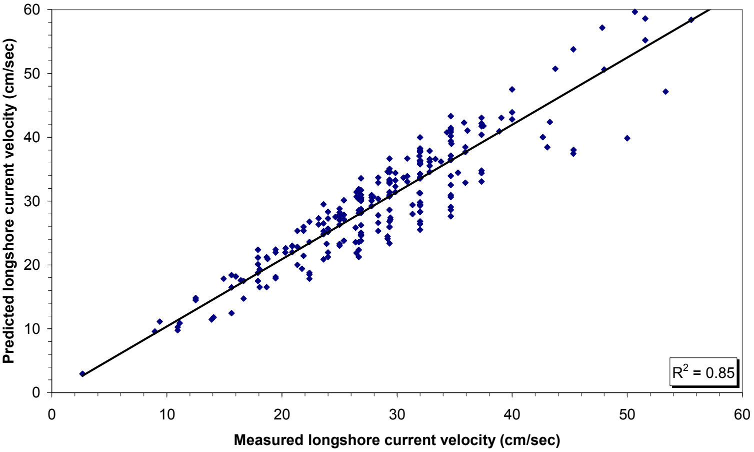


Figure 5: Scatter plot comparing observed and predicted daily longshore current velocity by ARIMA.

model can be seen to perform more satisfactorily than the ARIMA model. The CC scheme found that 58.27% and 15% decrease in RMSE over ARIMA and FF-BB, respectively.

Table 1 shows that forecasting accuracy decreased as the interval of forecasting reduced from one month to one day. As the noise in the time series increases, neural network performance generally gets worse. Similar to the monthly forecasting here also the neural network approach was more satisfactory than the traditional stochastic ARIMA model. Although both neural networks and ARIMA models perform worse as data becomes noisier, neural networks seem to be more robust as the differences between the two models are getting larger.

5 Conclusions

Longshore currents play the main role in sediment transport along the Egyptian Nile delta coast. The longshore current prediction in the conventional numerical models require a large amount information apart from historical wave observations and topography maps and are complex and tedious to apply specifically when point-forecasts at specific locations are needed. In this study, the neural mythology was implemented to forecast longshore current velocities. For this, the data were used, which were collected by Coastal Research Institute (CRI) for a period of sixteen years from 1982 to 1997 at Damietta promontory of Nile delta coast.

In this study two different NN training algorithms (PB and CC) and ARIMA were used in order to show the significance of suitable selection of the appropriate NN training algorithm.

A compression of results between feed forward neural network model (BP and CC) and the ARIMA model would show that the neural network perform much better than ARIMA. The four indices (MAPE, RMSE, SI, and R) were all the lowest for the neural network model. The results indicated that the ARIMA model does not capture the entire pattern of longshore current velocities. Among the PB sand CC networks employed for forecasting the longshore current velocities, the CC scheme was found to be superior to the PB scheme. The development of the longshore current velocities scheme in a model free manner seems to have resulted in more flexibility in the data mining approach of neural network than one associated with fixed type of stochastic time series schemes (ARIMA).

Forecasting accuracy decreased as the interval of forecasting reduced from one month to one day. This is expounded that the over fitting was made by large training patterns.

References

[1] El-Bisy, M.S., Longshore current prediction with neural networks. *Ain Shams Univ, Eng. Bulletim,* **40 (4)**, pp. 489-503, 2005.

[2] Thornton, E. B. & Guza, R.T., Surf zone longshore currents and random waves: field data and models. *Journal of Physical Oceanography*,**16**, pp.1165-1178,1986.

[3] Gourlay, M.R., Nonuniform alongshore currents. *Proc. 15th Coastal Engineering Conf.*, American Society of Civil Engineers, pp.701-720, 1976.

[4] Gourlay, M.R., *Nonuniform Alongshore Currents and Sediment Transport; A One-Dimensional Approach*. Research Report No. CE31, Department of Civil Engineering, University of Queensland, St. Lucia, Australia, 1982.

[5] Basco, D.R., *Surf Zone Currents. Miscellaneous*. Report No. 82-7 (I),US Army Engineer Waterways Experiment Station, Coastal Engineering Research Center, Vicksburg, MS, 1982.

[6] Basco, D.R. &Coleman, R.A., *Surf Zone Currents*. Miscellaneous Report No. 82-7 (II), US Army Engineer Waterways Experiment Station, Coastal Engineering Research Center, Vicksburg, MS, 1982.

[7] Deo, M.C. &Naidu, C.S., Real time wave forecasting using neural networks. *Ocean Engineering*, **26**, pp. 191–303, 1999.

[8] Box, G.E.P. & Jenkins, G.M., *Time Series Analysis Forecasting and Control*. Holden Day, San Francisco, CA, 1976.

[9] Zhang, G.P. & Qi, M., Neural network forecasting for seasonal and trend time series. *European J. of Operational Research*, **160**,pp. 501-514, 2005.

[10] Hoehfeld, M. & Fahlman, S. E., Learning with limited numerical precision using the cascade correlation learning algorithm. *IEEE Transactions on Neural Networks*, **3(4)**, pp. 602-611,1992.

[11] Adams, A. &Waugh, S., Function evaluation and the cascade correlation architecture. *ICNN*, Perth, Western Australia, pp. 1126- 1129, 1995.

[12] Wu, JK, *Neural Networks and Simulation Methods*. Marcel Dekker, New York, 1994.