

An Accurate Heave Signal Prediction Using Artificial Neural Network

Mohammed El-Diasty^{1,2}

¹Hydrographic Surveying Department, Faculty of Maritime Studies, King Abdulaziz University

²Engineering Department of Public Works, Faculty of Engineering, Mansoura University

Accepted 05 Oct 2014, Available online 15 Oct 2014, Vol.2 (Sept/Oct 2014 issue)

Abstract

An accurate heave modeling is required for several applications, including hydrographic surveying. This paper proposes an adaptive heave signal modeling, which uses a neural network-based modelling. A recurrent neural network and three-layer feed forward neural network trained using the Levenberg-Marquardt learning algorithm is used for this purpose. Computational results with five different datasets of real time heave are provided to validate the effectiveness of the artificial neural network-based model. It is shown that the new neural network-based model give a reliable heave model with excellent performance. Also, a comparison is made between the developed artificial neural network models and the autoregressive models. It is shown that the new artificial neural network-based model give the best performance results (i.e., the mean square error MSE).

Keywords: Heave Prediction, Neural Network, Autoregressive, Recurrent Network

1. Introduction

Accurate heave modeling is required for several applications, including hydrographic surveying. The physical models and Kalman Filter estimation of the heave process can be found in [1]. Frequency response methods have been used in the past to identify heave models. The model is based on the frequency content of the heave record. The frequency content is used as the basis to formulate a state space representation of the heave model [1]. A lot of studies have been reported to solve this problem, with the Kalman filter being the most common [1]. The design of the Kalman filter is based on the assumption of the complete structural knowledge of the model which describes the heave dynamics and it is based the Gaussian assumption of the state space noise statistics [2]. It is well known that there is a degradation of the KF-based estimation quality when the actual noise statistics is not Gaussian [3]. In order to overcome these limitations, another approach was followed based on autoregressive moving average model [4]. This paper proposes an adaptive heave signal modeling, which uses a neural network-based modelling. A recurrent neural network (RNN) and three-layer feed forward neural network (FFNN) trained using the Levenberg-Marquardt algorithm is used for this purpose. Computational results with five different datasets of real time heave are provided to validate the effectiveness of the ANN-based model. It is shown that the new artificial neural network-based model give a reliable heave model with excellent performance. Also, a comparison is made between the

developed artificial neural network models and the autoregressive models. It is shown that the new artificial neural network-based model give the best performance results (i.e., the mean square error MSE).

2. Autoregressive Models (AR)

Autoregressive process is linear model. The brief literature of autoregressive process can be found in [4] and [5]. Autoregressive (AR) process is time series consisting linear combination of immediate past values of the time series. Figure 1 shows an example of a fully connected AR structure, which is referred to as p-1 (p_AR, 1_out). AR(p) process of order p can be summarized in the following linear equation (see Figure 1):

$$\hat{y}(n) = \sum_{i=1}^p a(i).y(n-i) \quad (1)$$

Where, $\hat{y}(n)$ is the estimated output, $y(n-i)$ is the immediate observed heave value and $a(i)$ is autoregressive (AR) model parameter. AR(p) model parameters are estimated using least squares method [5]. To consider the non-stationary part (i.e., residuals) in the time series, the linear autoregressive moving average (ARMA) model. ARMA process is time series consisting of random error component and linear combination of immediate heave values and immediate past residuals. Figure 1 shows an example of a fully connected AR structure, which is referred to as (p+q)-1 (p_AR + q_MA,

1_out). The ARMA(p, q) process can be summarized in the linear equation (see Figure2):

$$y(n) = \sum_{i=1}^p a(i).y(n-i) + \sum_{j=1}^q c(j).e(n-j) + \varepsilon(n) \quad (2)$$

Where, $\hat{y}(n)$ is the estimated output, $y(n-i)$ is the immediate past value of the records, $a(i)$ is autoregressive (AR) model parameter, $e(n-j) = \hat{y}(n-j) - y(n-j)$ is immediate past residual, and $c(j)$ is moving average (MA) model parameter. ARMA(p, q) model parameters are estimated using least squares method [6].

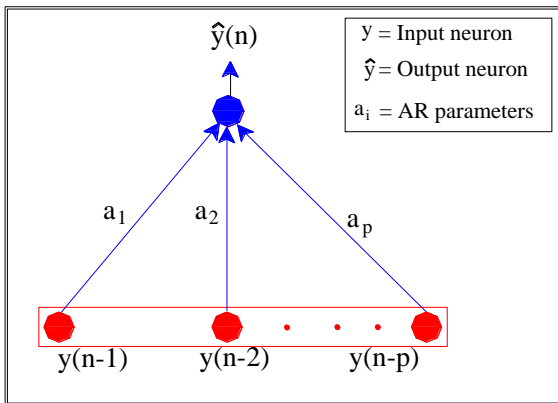


Fig. 1: Autoregressive (AR) structure [(p_AR)-(1_out)]

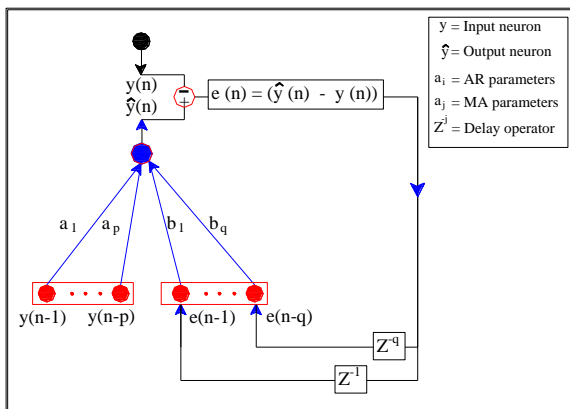


Fig. 2: Autoregressive Average Moving (ARMA) structure [(p_AR+q_MA)-(1_out)]

3. Artificial Neural Networks (ANN) Models

3.1 Basics of Artificial Neural Networks

In this section we briefly describe the fundamentals of artificial neural networks (for more details see [5]). Artificial neural network can be simply interpreted as a highly nonlinear autoregressive model which is constructed by combining nonlinear activation function through a multilayer structure. Neural network have been successfully applied in many different fields as a

prediction model (See [6]; [7]). Artificial neural networks can be developed in many different forms, depending on network structure and the learning algorithms. Figure 3 represents a block diagram of a simple model of a neuron showing the weights of the various links.

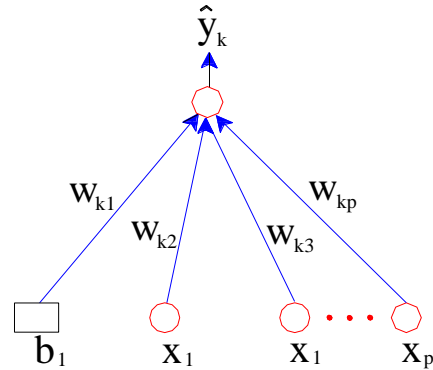


Fig. 3: Simple neural network structure

The above structure contains input layer and output layer and can be represented in the compact form as:

$$\hat{y}_k = f\left(\sum_{i=1}^p w_{ki}x_i + b_k\right) \quad (3)$$

Where, x_j represents one variable in the input layer, f is the activation function, \hat{y}_k represents the output variable, b_k is the bias variable, and w_{ki} represents one of the neural network parameter. Artificial neural networks are a flexible nonlinear form in which the activation functions f can be chosen arbitrary. Figure 4 show example of two common activation functions (i.e. Sigmoid and hyperbolic tangent function).

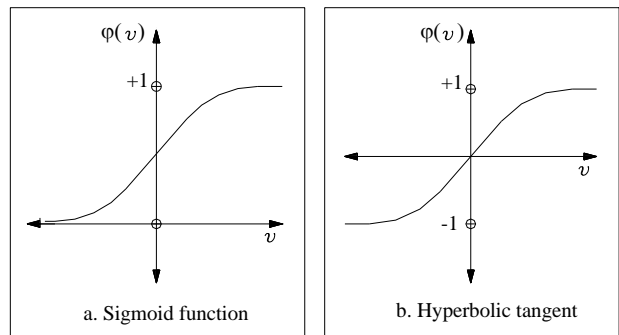


Fig. 4: Activation functions - sigmoid and hyperbolic tangent function.

In this paper we investigated the use of the feedforward neural networks and recurrent neural network in heave prediction model. It should be noted that for the prediction model, the input layer variable $y(n-i)$ represents the immediate past value of the time series records.

3.2 Feed forward and Recurrent Neural Network Heave Prediction Models

A feedforward networks are characterized by one input layer, one output layer and one or more hidden layers containing a number of network activated variables [5]. Figure 5 shows an example of a fully connected three-layer feedforward network in AR structure, which is referred to as which is referred to as [(p_AR)-(s_hidden)-(1_out)]. The input variables are simultaneously activated by a nonlinear activation function f_s and s provide hidden variables in a hidden layer. Then, the resulting hidden variables are activated by another activation function f_j and produce the network output J .

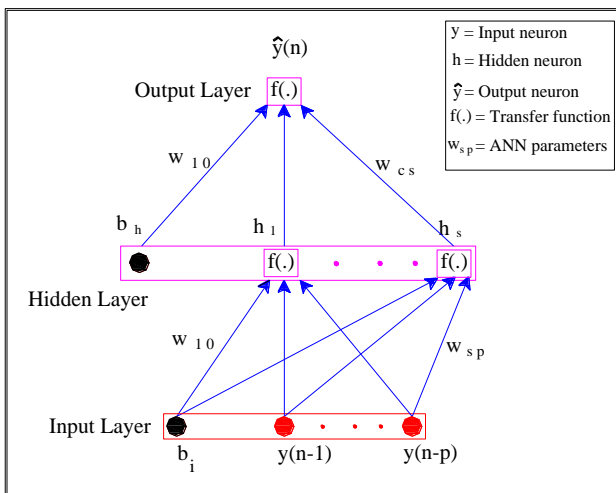


Fig. 5: Three-Layer Feedforward Neural Network with the Structure [(p_AR)-(s_hidden)-(1_out)].

The above structure can be represented in a compact form as:

$$\hat{y}(n) = f \left(\sum_{k=1}^h w_{nk}^y \cdot f_k \left(\sum_{i=1}^p w_{ki}^h \cdot y(n-i) + b_k \right) + b_n \right) \quad (4)$$

Alternatively, a recurrent neural network is a highly nonlinear dynamic structure and similar to ARMA model in the structure [9]. From a dynamic point of view, it is important to include the lagged dependent variables as feedback loops. The lagged variables are the residual errors between the desired outputs and the estimated outputs are considered as input variables. Hence, a recurrent neural network has been proposed in this paper. A recurrent neural networks are similar to the feedforward networks, with except that the former have at least one feedback loop). Figure 6 shows an example of a fully connected three-layer recurrent network in ARMA structure, which is referred to as [p_AR + q_MA)-(s_hidden)-(1_out)].

The following structure can be represented in a compact form as:

$$\hat{y}(n) = f \left(\sum_{k=1}^h w_{nk}^y \cdot f_k \left(\sum_{i=1}^p w_{ki}^h \cdot y(n-i) + \sum_{j=1}^q w_{kj}^h \cdot e(n-j) + b_k \right) + b_n \right) \quad (5)$$

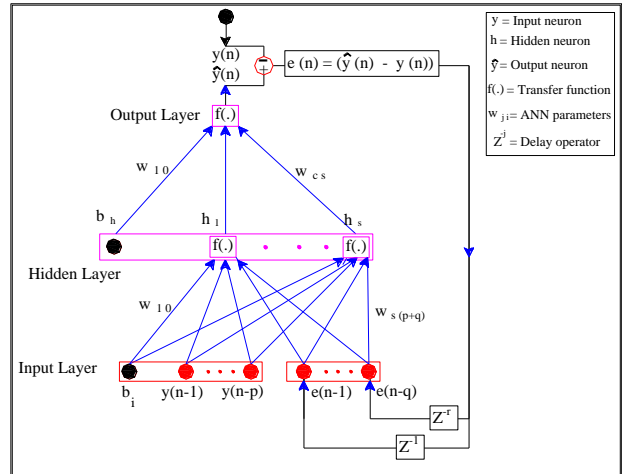


Fig. 6 Recurrent Neural Network structure [(p_AR + q_MA)-(s_hidden)-(1_out)]

Training the feed forward neural network and recurrent neural network is accomplished through iterative adjustments of the free parameters, i.e., the weights and bias, of the network till we obtain the optimal values. There exist various learning algorithms, which are fundamental to the design of neural networks. The Levenberg-Marquardt learning algorithm is the most widely used for feedforward neural networks and recurrent neural network, which is used in this research (see [9] for more details).

4. Results and discussion

Five different data series of real time heave, namely MBheave-1, MBheave-2, MBheave-3, MBheave-4, and MBheave-4, were used to validate the neural network model. The structure of the neural network was built using the Matlab [8]. Several tests were conducted to optimize the structure of the neural network for each of the five data series. As mentioned before, we have investigated two neural network-based models. First, we used the feedforward neural network structure. In this approach, immediate past values of the heave records are used as input to the network, while future values of the tidal records are used as the desired output. Then, we followed another neural network structure, which is recurrent neural network. In this approach, the immediate past values of the heave records and the immediate past of value of the residuals are used as input to the network, while future values of the heave records are used as the desired output.

Each of the data series was divided into three datasets: training, testing and validation datasets. The first 10% heave records were assigned to the testing dataset, while the last 10% heave records were assigned to the validation dataset. The training dataset was

Table 1: Summary results

Data	Model	AR	ARMA	FFNN	RNN
MB-1	MSE-train	1.13E-05	1.07E-05	5.62E-06	5.07E-06
	MSE-valid	1.88E-05	1.85E-05	9.53E-06	8.17E-06
MB-2	MSE-train	7.53E-06	7.58E-06	3.77E-06	3.61E-06
	MSE-valid	5.89E-06	5.79E-06	2.96E-06	2.83E-06
MB-3	MSE-train	1.34E-05	1.18E-05	6.58E-06	5.71E-06
	MSE-valid	1.01E-05	8.62E-06	5.13E-06	4.60E-06
MB-4	MSE-train	1.55E-05	1.35E-05	7.77E-06	6.81E-06
	MSE-valid	1.14E-05	1.04E-05	5.72E-06	5.37E-06
MB-5	MSE-train	1.52E-05	1.34E-05	7.59E-06	6.64E-06
	MSE-valid	1.99E-05	1.75E-05	1.00E-05	8.69E-06

selected to represent the middle portion of the data series, which varied from one MBheave to another. The training was stopped based on testing the generalization performance the neural network using the testing dataset. After training and testing the network, we generalized the model to predict ahead the last 10% values of the data series and compared the results with the observed heave records. This was done in a sequential manner to emulate the real-time condition. It was concluded that the recurrent neural network in ARMA structure [(6_AR+ 6_MA)-(12_hidden)-(1_out)] gives the best results, i.e., has the lowest MSE (MSE) error (See Table 1).

To further validate our model, we compared it with the autoregressive approach. Autoregressive (AR) structure and autoregressive Moving Average (ARMA) structure were built using the Matlab [8]. Each of the data series was divided into two datasets: training, and validation datasets. The first 90 % heave records were assigned to the training dataset, while the last 10% heave records were assigned to the validation dataset. Several tests were conducted to optimize the structure of the AR and ARMA for each of the five data series. It was concluded that the ARMA with the structure [6_AR + 6_MA)-(1_out)] gives the best results, i.e., has the lowest MSE (MSE) error.

results were obtained for the other four data series. It can be seen that the maximum prediction error is approximately 1 cm, with the bulk of the residuals fall within the +/-0.5 cm range (see Figure 7). This shows that the developed neural network model has the capability to precisely predict the heave values.

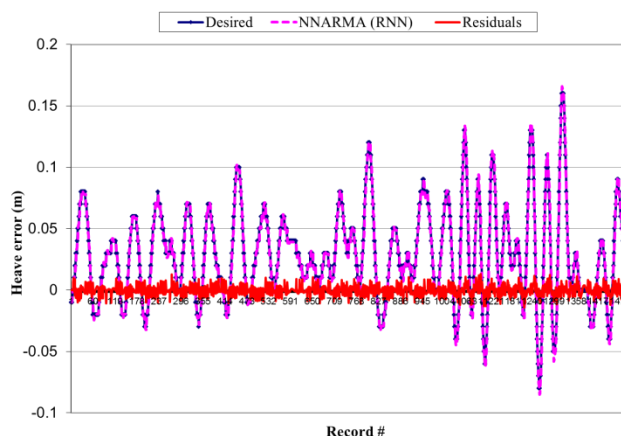


Fig. 8: Predicted heave versus the desired (i.e., actual) values, and the prediction error using Artificial Neural Network (RNN)

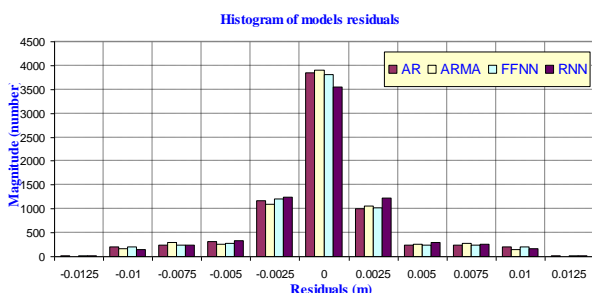


Fig. 7: Histogram for all models' residuals

The overall conclusion from Table 1 and Figure 7 was that the recurrent neural network with the structure [6_AR + 6_MA)-(12_hidden)-(1_out)] gives the best results, i.e., has the lowest MSE (MSE) error. Figures 8 shows that the predicted heave versus the desired values, and the prediction errors (i.e., residuals) for MBheave-1. Similar

Conclusion

A sequential heave prediction model using Artificial Neural Networks was developed in this paper. The recurrent neural network structure gave the best performance results (i.e., the minimum RMS), and therefore was used in predicting the heave values. Heave data series, with varying lengths, from five different real time heave data series were used to verify the model. It is shown that the maximum prediction error is approximately 1 cm, with the bulk of the residuals fall within the +/-0.5 cm range. A performance comparison was made between the developed neural network model and the autoregressive method for heave prediction. It is shown that performance of recurrent neural network is better than autoregressive method i.e., has the lowest MSE (MSE) error. Future development will include a complete attitude and heave solution, which takes advantage of GNSS heights

References

- [1]. El-Hawary, F. (1982): "Compensation for Source Heave by Use of a Kalman Filtering", IEEE, Journal of oceanic Engineering, Vol. OE-7, No. 2.
- [2]. Brown R. G., and P. Hwang (1997): "Introduction to Random Signals and Applied Kalman Filtering." John Wiley & Sons, Inc., Toronto, Canada.
- [3]. Kai-Wei Chiang, Aboelmagd Noureldin, and Naser El-Sheimy (2003): "Multi-sensors Integration using Neuron Computing for Land Vehicle Navigation", Journal of "GPS Solutions" Vol 6, No. 4, pp. 209-218, Published by John Wiley & Sons, Inc.
- [4]. El-Hawary, F., G. Mbamalu (1993): "Refined Autoregressive Moving –Average Modeling of Underwater Heave Process." IEEE, Journal of Oceanic Engineering, Vol. 18, No. 3.
- [5]. Haykin, S. (1999): Neural Networks: A Comprehensive Foundation. Second Edition. Prentice Hall.
- [6]. El-Rabbany, A., G. Auda and S. Abdelazim (2002). "Predicting Sea Ice Conditions Using Neural Networks." Journal of Navigation, Vol. 55, Number 1, pp. 137-143.
- [7]. El-Rabbany, A. and M. El-Diasty (2003). "A New Approach to Sequential Tidal Prediction." The Journal of Navigation, Vol. 56, pp. 305-314.
- [8]. MathWorks (2004). "Matlab 6.5 Reference Guide", MathWorks Inc., USA.
- [9]. Nørgaard, M., O. Ravn, N. K. Poulsen, L. K. Hansen (2000): "Neural networks for Modelling and Control of Dynamic Systems," Springer-Verlag, London, UK.