KSCE Journal of Civil Engineering (2015) 19(2):401-417
DOI 10.1007/s12205-015-1483-9

Hydro-Climatological Analysis for Floods and Droughts

pISSN 1226-7988, eISSN 1976-3808
www.springer.com/12205

Multistep-Ahead Flood Forecasting Using Wavelet and Data-Driven Methods

Youngmin Seo*, Sungwon Kim, and Vijay P. Singh*****

Received August 22, 2014/Revised November 11, 2014/Accepted December 4 2014

Abstract

Accurate forecasting of floods is vital for developing a flood warning systems, flood prevention, flood damage mitigation, soil erosion reduction and soil conservation. The objective of this study is to apply two hybrid models for flood forecasting and investigate their accuracy for different lead times. These two models are the Wavelet-based Artificial Neural Network (WANN) and the Wavelet-based Adaptive Neuro-Fuzzy Inference System (WANFIS). Wavelet decomposition is employed to decompose the flood time series into approximation and detail components. These decomposed time series are then used as inputs of Artificial Neural Network (ANN) and adaptive Neuro-Fuzzy Inference System (ANFIS) modules in the WANN and WANFIS models, respectively. The WANN and WANFIS models yielded better results than the ANN and ANFIS models for different lead times. The WANN and WANFIS models performed almost similarly. However, in terms of model efficiency, the WANFIS model was superior to other models for lead times of 1 to 6 hours, and the WANN model was superior to other models for lead time of 8 to 10 hours. The results obtained from this study indicate that the combination of wavelet decomposition and data-driven models, including ANN and ANFIS, can improve the efficiency of data-driven models. Results also indicate that the combination of wavelet decomposition and data-driven models can be a potential tool for forecasting flood stage more accurately.

Keywords: *flood stage forecasting, discrete wavelet decomposition, data-driven methods, artificial neural network, adaptive neuro-fuzzy inference system, Wavelet-based ANN, Wavelet-based ANFIS*

1. Introduction

Climate change has been alleged to increase extreme hydrological events, including heavy rainfalls and floods. In South Korea, weather conditions due to mountainous terrain further exacerbate these extreme events. Heavy rainfall, especially during a typhoon event, causes severe flood damages in the downstream region. In addition to the geographical and hydrological characteristics, overpopulation, urbanization and industrialization further complicate flood control and management in South Korea.

Due to the spatial and temporal variation of the rainfall distribution and the inordinately complex and highly nonlinear nature of rainfall-runoff relationship, flood forecasting remains one of the most challenging and important tasks of operational hydrology (Chang *et al.*, 2007). Accurate forecasting of the flood time series is vital for developing flood warning systems, flood prevention, flood damage mitigation, soil erosion reduction and soil conservation. Flood forecasting represents a complex nonlinear problem and hence it is difficult to model.

For decades, many studies have been carried out for flood forecasting based on conceptual and stochastic models (Salas *et al.*, 1985; Brath and Rosso, 1993; Chang and Hwang, 1999; Toth *et al.*, 1999; Mishra *et al.*, 2004; Calvo and Savi, 2009). Although conceptual models are reliable in forecasting the important hydrograph features, implementation and calibration of such models can have several difficulties, including development of sophisticated mathematical tools, estimation of many parameters for modeling and some degree of modeling experience (Duan *et al.*, 1992; Grayson *et al.*, 1992). Stochastic approaches to flood forecasting have been undertaken using linear models, including autoregressive (AR), Autoregressive Moving Average (ARMA), autoregressive moving average with exogenous inputs (ARMAX), and nonlinear regression models (Salas *et al.*, 1985; Chang and Hwang, 1999; Mishra *et al.*, 2004). Applicability of these models can however be limited to basins with streamflow data measured for long periods and no significant change in watershed conditions (Baratti *et al.*, 2003).

Over the past years, data-driven methods have been successfully developed for modeling non-linear hydrologic systems. Especially, Artificial Neural Network (ANN) and Adaptive Neuro-fuzzy

*Member, Adjunct Professor, Dept. of Constructional Disaster Prevention Engineering, Kyungpook National University, Sangju 742-711, Korea (E-mail: ymseo.hydro@gmail.com)
**Member, Associate Professor, Dept. of Railroad and Civil Engineering, Dongyang University, Yeongju 750-711, Korea (Corresponding Author, E-mail: swkim1968@dyu.ac.kr)
***Distinguished Professor and Caroline & William N. Lehrer Distinguished Chair in Water Engineering, Dept. of Biological and Agricultural Engineering & Zachry Dept. of Civil Engineering, Texas A & M University, College Station, Texas 77843-2117, USA (E-mail: vsingh@tamu.edu)

Inference System (ANFIS) have been accepted as effective tools for flood forecasting (Campolo *et al.*, 2003; Nayak *et al*., 2005; Piotrowski *et al.*, 2006; Chang *et al.*, 2007; Chiang *et al*., 2007; Kashani e*t al*., 2007; Kim and Kim, 2008; Mukerji *et al.*, 2009; Deshmukh and Ghatol, 2010; Tiwari and Chatterjee, 2010a; Nguyen and Chua, 2012; Seo *et al.*, 2013a, 2013b; Patel and Parekh, 2014; Rezaeianzadeh *et al.*, 2014). The ANN is parallel computational models that resemble biological neural network and have better generalization capabilities. The ANFIS, on the other hand, combines the advantages of both the ANN and fuzzy inference system (Okkan, 2012). Although ANN and ANFIS have been used extensively for forecasting hydrological variables, including streamflow, groundwater, and evaporation, they have some problems when it comes to deal with non-stationary and extreme value data. Since a hydrological time series includes several frequency components and has nonlinear relationships, hybrid model approaches, which include different data-preprocessing and combine techniques, have been found to improve the performance of forecasting models (Okkan, 2012).

Recently, the combination of wavelet transform and data-driven models have been successfully applied for flood forecasting (Adamowski, 2008; Tiwari and Chatterjee, 2010b; Dadu and Deka, 2013; Sehgal *et al.*, 2014). The wavelet transform is a data-preprocessing technique that can analyze a signal in both time and frequency so that it can overcome the drawbacks of the conventional Fourier transform. The wavelet transform provides effective decomposition of time series so that decomposed data can increase the performance of hydrological prediction models by capturing useful information on different resolution levels (Nourani *et al.,* 2009; 2011).

Adamowski (2008) developed a method of stand-alone short-term spring snowmelt river flood forecasting based on wavelet and cross-wavelet analysis. The wavelet forecasting method was compared to Multiple Linear Regression (MLR), Autoregressive Integrated Moving Average (ARIMA), and Artificial Neural Network (ANN) analysis for forecasting daily stream flow with lead times equal to 1, 2, and 6 days. The author found that the best model for lead times of 1 and 2 days was the wavelet forecasting model and the best model for lead time of 6 days was the ANN model. The author also found that the wavelet forecasting model was not particularly accurate for longer lead time forecasting such as 6 days.

Tiwari and Chatterjee (2010b) developed a hybrid Wavelet-Bootstrap-ANN (WBANN) model to explore the potential of wavelet and bootstrapping techniques for developing an accurate and reliable ANN model for hourly flood forecasting. The performance of WBANN model was compared with three different ANN models, including traditional ANN, wavelet-based ANN (WANN), and bootstrap-based ANN (BANN). They found that the overall performance of WBANN model was accurate and reliable as compared to the other three models. They revealed that wavelet decomposition improved the performance of ANN models and the bootstrap resampling technique produced more consistent and stable solutions.

Dadu and Deka (2013) developed a river flow forecasting model based on Wavelet And Artificial Neural Network (WANN) methods. They found that the WANN model gave better and more consistent results than the ANN model for almost all lead times. They also found that the WANN model with *db*5 mother wavelet gave slightly better results for all lead times than the WANN model with *db*4 mother wavelet.

Sehgal *et al.* (2014) developed two types of Wavelet-Based Adaptive Neuro-Fuzzy Inference System (WANFIS), including WANFIS-split data model (WANFIS-SD) and WANFIS-modified time series model (WANFIS-MS) for forecasting river water levels with a 1-day lead time. They found that the proposed models forecasted river water levels accurately and the WANFIS-SD performed better than the WANFIS-MS for high flood levels.

This study applies two hybrid models for flood forecasting and investigates their accuracy for different lead times. These two hybrid models are the Wavelet-based Artificial Neural Network (WANN) and the Wavelet-based Adaptive Neuro-Fuzzy Inference System (WANFIS). The model performance is evaluated for flood forecasting in Bocheong stream catchment, South Korea, using standard goodness-of-fit measures. The performance of the WANN and WANFIS models is compared with that of the traditional ANN and ANFIS models for different lead times.

2. Methodology

2.1 Wavelet Decomposition

Wavelet analysis is a multi-resolution analysis in time and frequency domain. The wavelet transform decomposes the time series signal into different resolutions by controlling scaling and shifting. It provides good localization properties obtained in both time and frequency domains (Nejad and Nourani, 2012).

The Continuous Wavelet Transform (CWT) of a signal $x(t)$ is defined as (Adamowski and Sun, 2010):

$$CWT_x^{\Psi}(\tau,s) = \frac{1}{\sqrt{|s|}}\int_{-\infty}^{+\infty} x(t)\Psi^*\left(\frac{t-\tau}{s}\right)dt \tag{1}$$

where s is the scale parameter, τ is the translation parameter, '*' denotes the complex conjugate, and $\Psi(t)$ is the mother wavelet. The CWT necessitates a large amount of computation time and resources, while Discrete Wavelet Transform (DWT) requires less computation time and is simpler to implement than the CWT. The DWT involves choosing scales and positions, which are called dyadic scales and positions, based on the power of two. This is achieved by modifying the wavelet representation as follows (Adamowski and Sun, 2010):

$$\Psi_{j,k}(t) = \frac{1}{\sqrt{|s_0^j|}}\Psi\left(\frac{t-k\tau_0 s_0^j}{s_0^j}\right) \tag{2}$$

where j and k are integers that control the wavelet dilation and translation, respectively, $s_0 > 1$ is a fixed dilation step, and τ_0 is the location parameter. The most common and simplest choices

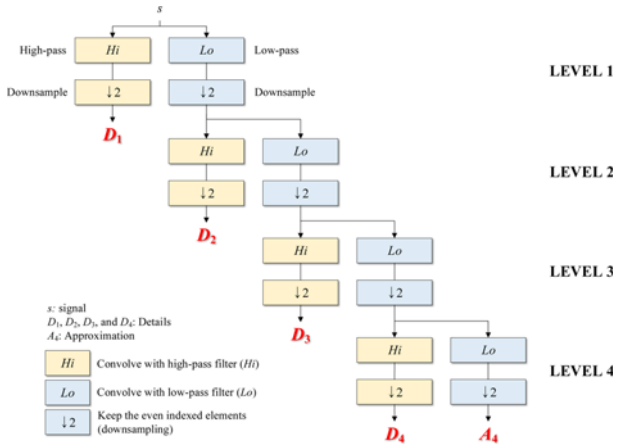

Fig. 1. Mallat's Algorithm Demonstration for Four-Level Decomposition of a Signal

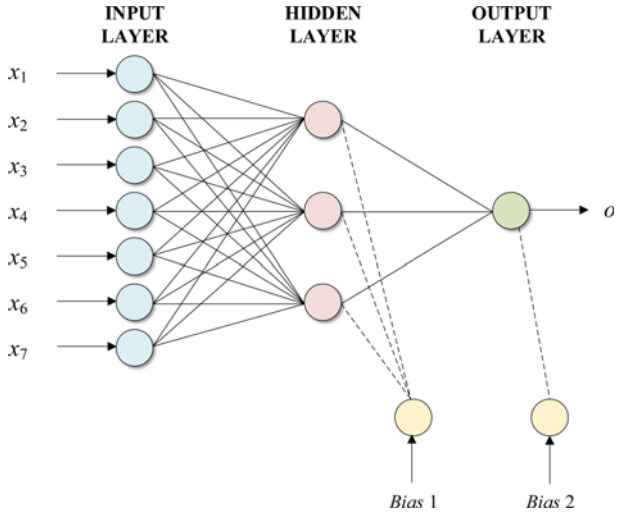

Fig. 2. General Architecture of ANN

for the parameters are: $s_0 = 2$ and $\tau_0 = 1$ (Nourani *et al.*, 2009). By the wavelet discretization, the time-scale space can now be sampled at discrete levels.

A fast DWT algorithm developed by Mallat (1989) is based on four filters, including decomposition low-pass, decomposition high-pass, reconstruction low-pass and reconstruction high-pass filters. The multi-resolution analysis by Mallat's algorithm is a procedure to obtain 'approximations' and 'details' from a given signal. An approximation holds the general trend of the original signal, whereas a detail depicts high-frequency components of it. By decomposing the approximations successively as shown in Fig. 1, a multi-level decomposition process can be achieved, where the original signal is broken down into lower resolution components (Catalão *et al.*, 2011). Detailed information for Mallat's algorithm can be found in Nason (2010).

2.2 Artificial Neural Network

ANN is parallel computing systems that were developed originally based on the structure and functional aspects of biological neural networks. Feed-forward ANN comprises a system of units, analogous to neurons that are arranged in layers (Imrie *et al.*, 2000). Multilayer Perceptron (MLP) is the most popular neural network architecture. MLP is typically composed of several layers of nodes (neurons). An input layer, which is the first layer, receives external information. The problem solution is obtained in an output layer which is the last layer. One or more intermediate layers, which are called hidden layers, separate the input and output layers. The nodes in adjacent layers are usually fully connected by acyclic arcs, which are called synapses, from the input layer to the output layer (Zhang *et al.*, 1998). Fig. 2 shows a general ANN architecture.

To each of the synapses, a weight is attached indicating the effect of corresponding neurons, and all data pass the neural network as signals. The signals are processed first by the so-called integration function combining all incoming signals and second by the so-called activation function transforming the output of the neuron (Günther and Fritsch, 2010).

The simplest MLP consists of an input layer with n covariates and an output layer with one output neuron. It calculates the following function (Günther and Fritsch, 2010):

$$o(\mathbf{x}) = f\left(w_0 + \sum_{i=1}^{n} w_i x_i\right) = f(w_0 + \mathbf{w}^T \mathbf{x}) \tag{3}$$

where w_0 is the intercept, $\mathbf{w} = (w_1, \ldots, w_n)$ is the vector consisting of all synaptic weights without the intercept, and $\mathbf{x} = (x_1, \ldots, x_n)$ is the vector of all covariates.

Hidden layers can be included to increase the flexibility of the model. Hornik *et al.* (1989) showed that one hidden layer is sufficient to model any piecewise continuous function. The MLP with a hidden layer and J hidden neurons calculates the following function (Günther and Fritsch, 2010):

$$\begin{aligned} o(\mathbf{x}) &= f\left(w_0 + \sum_{j=1}^{J} w_j \cdot f\left(w_{0j} + \sum_{i=1}^{n} w_{ij} x_i\right)\right) \\ &= f\left(w_0 + \sum_{j=1}^{J} w_j \cdot f(w_0 + \mathbf{w}_j^T \mathbf{x})\right) \end{aligned} \tag{4}$$

where w_{0j} is the intercept of the jth hidden neuron, w_j is the synaptic weight corresponding to the synapse starting at the jth hidden neuron and leading to the output neuron, $\mathbf{w}_j = (w_1, \ldots, w_{nj})$ is the vector of all synaptic weights corresponding to the synapses leading to the jth hidden neuron, and $\mathbf{x} = (x_1, \ldots, x_n)$ is the vector of all covariants.

All hidden neurons and output neurons calculate an output of $f(g(z_0, z_1, \ldots, z_k)) = f(g(\mathbf{z}))$ from the outputs of all preceding neurons, $z_0, z_1, \ldots, z_k$, where g is the integration function, f is the activation function, and the neuron $z_0 \equiv 1$ is the intercept. The integration function is often defined as $g(\mathbf{z}) = w_0 z_0 + \sum_{i=1}^{k} w_i z_i = w_0 + \mathbf{w}^T \mathbf{z}$. The activation function is usually a bounded non-decreasing nonlinear and differentiable function, including the logistic function or the hyperbolic tangent (Günther and Fritsch, 2010).

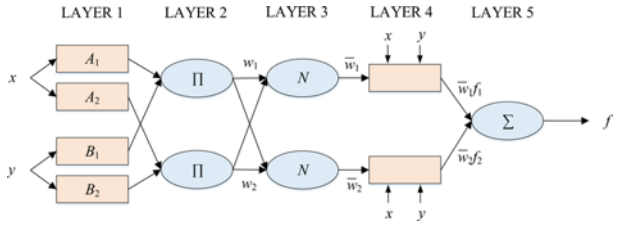

Fig. 3. General Architecture of ANFIS

2.3 Adaptive Neuro-Fuzzy Inference System

ANFIS is a combination of an adaptive neural network and a Fuzzy Inference System (FIS). Since this system is based on the FIS reflecting vague knowledge, an important aspect is that the system should always be interpretable in terms of fuzzy IF-THEN rules. The ANFIS can approximate any real continuous function on a compact set to any degree of accuracy (Jang *et al.*, 1997). There are two approaches for FIS, namely the approaches of Mamdani (Mamdani and Assilian, 1975) and Sugeno (Takagi and Sugeno, 1985). The differences between the two approaches arise from the consequent part. Mamdani's approach uses fuzzy Membership Functions (MFs), whereas Sugeno's approach uses linear or constant functions. In this study, Sugeno's approach was used for flood forecasting. Fig. 3 shows the general ANFIS structure.

The procedure described by Jang *et al.* (1997) was adopted for flood forecasting in this study. As a simple example of the procedure adopted, a FIS with two inputs and one output is considered. The ANFIS has five layers comprising different node functions, as shown in Fig. 3. The output of the *i*th node in layer 1 is denoted as $O_{1,i}$. Every node *i* in layer 1 is an adaptive node, with $O_{1,i} = \varphi_{A_i(x)}$ for i = 1, 2 or $O_{1,i} = \varphi_{B_{i-2}}(y)$ for i=3, 4, where x (or y) is the input to node i, and A_i (or B_{i-2}) is a linguistic label (e.g., LOW or HIGH) associated with this node. The Gaussian MFs for A and B can be written as follows:

$$\varphi_{A_i}(x) = \exp\left[\frac{-(x-c)^2}{2\sigma^2}\right] \text{ and } \varphi_{B_{i-2}}(y) = \exp\left[\frac{-(y-c)^2}{2\sigma^2}\right] \quad (5)$$

where $[\sigma, c]$ is the parameter set. Parameters in layer 1 are called premise parameters. Any continuous and piecewise differential functions, such as triangular-shaped or bell-shaped MFs, are also qualified candidates for the node function in layer 1 (Jang, 1993).

Layer 2 consists of the nodes labeled by Π, which multiplies the incoming signals and sends the product out. The output of layer 2 comprises the membership values of the premise part and can be written as follows:

$$Q_{2,i} = w_i = \varphi_{A_i}(x)\varphi_{B_i}(y), \; i = 1,2 \quad (6)$$

The nodes labeled by N calculate the ratio of the ith rule's firing strength to the sum of all rules' firing strengths in layer 3. Each node output represents the firing strength of a rule and can be written as follows:

$$Q_{3,i} = \overline{w}_i = \frac{w_i}{w_1 + w_2}, \; i = 1,2 \quad (7)$$

where $\overline{w}_i$ is the output of layer 3. The outputs of layer 3 are called normalized firing strengths. The nodes of layer 4 are adaptive with node functions and can be written as follows:

$$Q_{4,i} = \overline{w}_i f_i = \overline{w}_i(p_i x + q_i y + r_i), \; i = 1,2 \quad (8)$$

where $[p_i, q_i, r_i]$ is the parameter set. The parameters of layer 4 are referred to as consequent parameters. The single fixed node of layer 5 labeled by Σ computes the final output as the summation of all incoming signals which can be written as follows:

$$O_{5,i} = \sum_{i=1}^{2} \overline{w}_i f_i = \frac{\sum_{i=1}^{2} w_i f_i}{\sum_{i=1}^{2} w_i} \quad (9)$$

The ANFIS is trained using a hybrid learning algorithm, combining the least-squares method and the backpropagation gradient descent method, to adjust the premise and consequent parameters. In forward pass, the least-squares method is used to identify consequent parameters. In backward pass, the gradient descent method is used to propagate the errors backward and adjust the premise parameters. Detailed information for ANFIS can be found in Jang (1993).

2.4 WANN and WANFIS

The WANN and WANFIS models are hybrid models combining wavelet decomposition and data-driven models. The WANN model is the conjunction model of wavelet decomposition and ANN model, whereas The WANFIS model is the conjunction model of wavelet decomposition and ANFIS model. The WANN and WANFIS models consist of a two-step algorithm as follows:

1. The first step is multi-level wavelet decomposition. The original time series are decomposed using wavelet transform after determining the decomposition level.
2. The second step is training and testing phases using ANN and ANFIS models. The details and approximation, which are obtained by the wavelet transform in the first step, are used as input of the ANN and ANFIS models.

In this study, DWT was used for decomposing the flood time series. Fig. 4 shows a flowchart for flood forecasting using the WANN and WANFIS models.

2.5 Performance Evaluation

The performance of flood forecasting models, including ANN, ANFIS, WANN and WANFIS models, was evaluated using seven performance indexes, including the Coefficient of Efficiency (CE), the index of agreement (d), the coefficient of determination (r^2), the Root-Mean-Square Error (RMSE), the Mean Absolute Error (MAE), the Mean Squared Error (MSE) and the Mean Squared Relative Error (MSRE). Performance indexes, CE, d, r^2, RMSE, MAE, MSE and MSRE, can be written as follows (Dawson and Wilby, 2001):

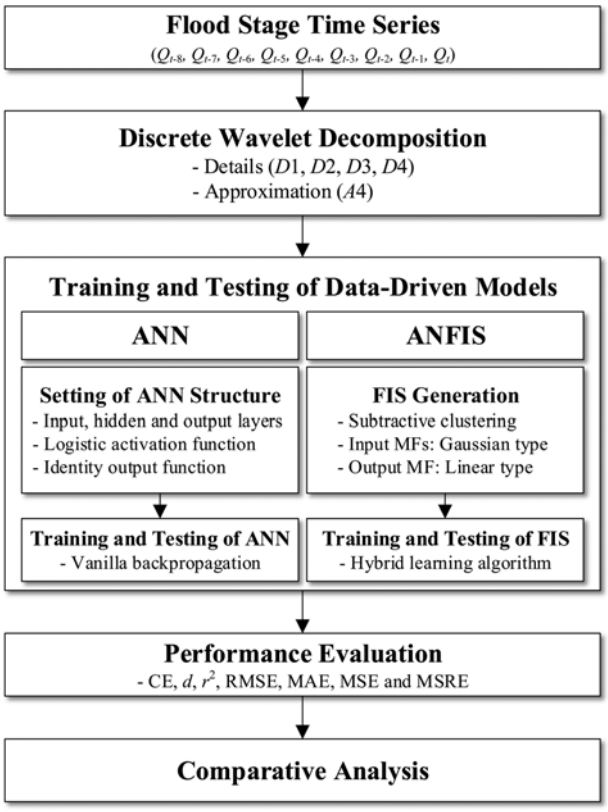

Fig. 4 Flowchart for WANN and WANFIS

$$CE = 1 - \frac{\sum_{i=1}^{N}(Q_i - Q_i^*)^2}{\sum_{i=1}^{N}(Q - \overline{Q})^2} \quad (10)$$

$$d = 1 - \frac{\sum_{i=1}^{N^2}(Q_i - Q_i^*)^2}{\sum_{i=1}^{N}(|Q_i^* - \overline{Q}| + |Q_i - \overline{Q}|)^2} \quad (11)$$

$$r^2 = \left| \frac{\sum_{i=1}^{N}(Q_i - \overline{Q})(Q_i^* - \tilde{Q})}{\sqrt{\sum_{i=1}^{n}(Q_i - \overline{Q})^2 \sum_{i=1}^{N}(Q_i^* - \tilde{Q})^2}} \right|^2 \quad (12)$$

$$RMSE = \left\{ \frac{1}{N} \sum_{i=1}^{N} [Q_i^* - Q_i]^2 \right\}^{0.5} \quad (13)$$

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |Q_i^* - Q_i| \quad (14)$$

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (Q_i - Q_i^*)^2 \quad (15)$$

$$MSRE = \frac{1}{N} \sum_{i=1}^{N} \frac{(Q_i - Q_i^*)^2}{Q_i^2} \quad (16)$$

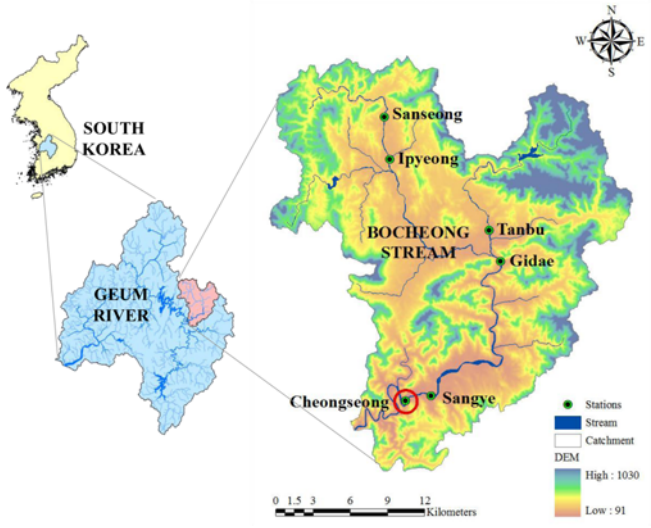

Fig. 5. Study Region and Locations of Stage Gauging Stations

where Q_i^* is the forecasted values, Q_i is the observed values, $\overline{Q}$ is the mean of the observed values, $\tilde{Q}$ is the mean of the forecasted values, and N is the number of observed values.

3. Case Study

The Bocheong stream catchment in South Korea was chosen for applying flood forecasting models in this study. As shown in Fig. 5, the catchment is located in the Geum River basin which is in the western region of South Korea. The Bocheong stream has a catchment area of 553.34 km^2 and a stream length of 68.04 km. The catchment has an average elevation of 263.9 m and an average slope of 32.09%.

Hourly stage data from a streamflow gauging station, Cheongseong, were obtained from the observation archive of Water Management Information System (WAMIS), which is operated by the Ministry of Land, Infrastructure and Transport (MOLIT), South Korea. Fig. 5 shows the locations of streamflow gauging stations in the catchment. The collected hourly stage data were prepared from 1 June to 30 September for a period between 2006 and 2010. The data were divided into two parts, data of the first four years for model training and the remaining data for model testing.

4. Application and Results

4.1 Analysis

For applying ANN, ANFIS, WANN and WANFIS models for flood forecasting, appropriate input variables must be selected in advance. This study used a statistical approach suggested by Sudheer *et al.* (2002) to identify the appropriate input variables. The method is based on the heuristic that the potential variables corresponding to different lag times can be identified using statistical analysis, including Cross Correlation Function (CCF), Autocorrelation Function (ACF) and Partial Autocorrelation

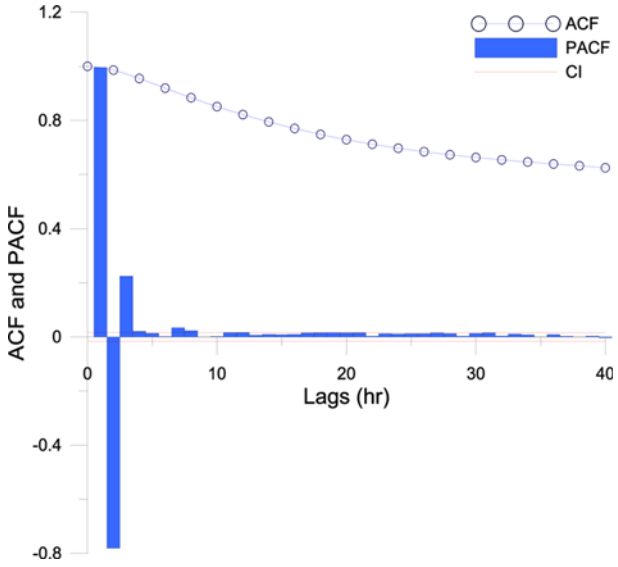

Fig. 6. ACF and PACF for Input Structure Identification

Function (PACF). Fig. 6 shows ACF and PACF for flood stage time series used in this study. The ACF, PACF and 95% confidence band were estimated from 1 to 20 hour lags. The PACF showed a significant correlation up to 9 hour lags and, thereafter, fell within the confidence band. The analysis suggested incorporating the flood stage values up to 9 hour lags in the input variables. The structure for the ANN and ANFIS models can be written as follows:

$$Q_{t+i} = f(Q_{t-8}, Q_{t-7}, Q_{t-6}, Q_{t-5}, Q_{t-4}, Q_{t-3}, Q_{t-2}, Q_{t-1}, Q_t) \quad (17)$$

where Q_{t-j} are input variables lagged by j hour ($j = 0, 1, \ldots, 8$) and Q_{t+i} are output variables for lead time i hour ($i = 1, 2, \ldots, 10$).

This study used DWT for decomposing the flood stage time series into wavelet components. The optimal decomposition level must be selected beforehand to determine the performance of the model in the wavelet domain. Many researchers have used an empirical equation to determine the decomposition level (Nourani *et al.*, 2009; Tiwari and Chatterjee, 2010b; Adamowski and Chan, 2011; Belayneh and Adamowski, 2012; Nejad and Nourani, 2012). In this study, the decomposition level was determined using the following empirical equation (Nourani *et al.*, 2009):

$$L = \text{int}[\log(N)] \quad (18)$$

where L is the decomposition level, N is the number of time series data, and int[·] is the integer-part function. In this study, four decomposition levels ($L = 4$) were determined using Eq. (18). The flood stage times series were decomposed using Daubechies-10 mother wavelet (*db*10) and DWT, and sub-time series of 2-hour mode ($D1$), 4-hour mode ($D2$), 8-hour mode ($D3$), 16-hour mode ($D4$); and the approximation modes ($A4$) were obtained for the training and testing periods. Fig. 7 shows flood stage time series and sub-time series. Since the flood stage time series, Q_{t-j} ($j = 0, 1, \ldots, 8$), was decomposed into five sub-time series components, $D1$, $D2$, $D3$, $D4$ and $A4$, a total of 45 sub-time series were used as inputs for WANN and WANFIS models. The structure for the WANN and WANFIS models can

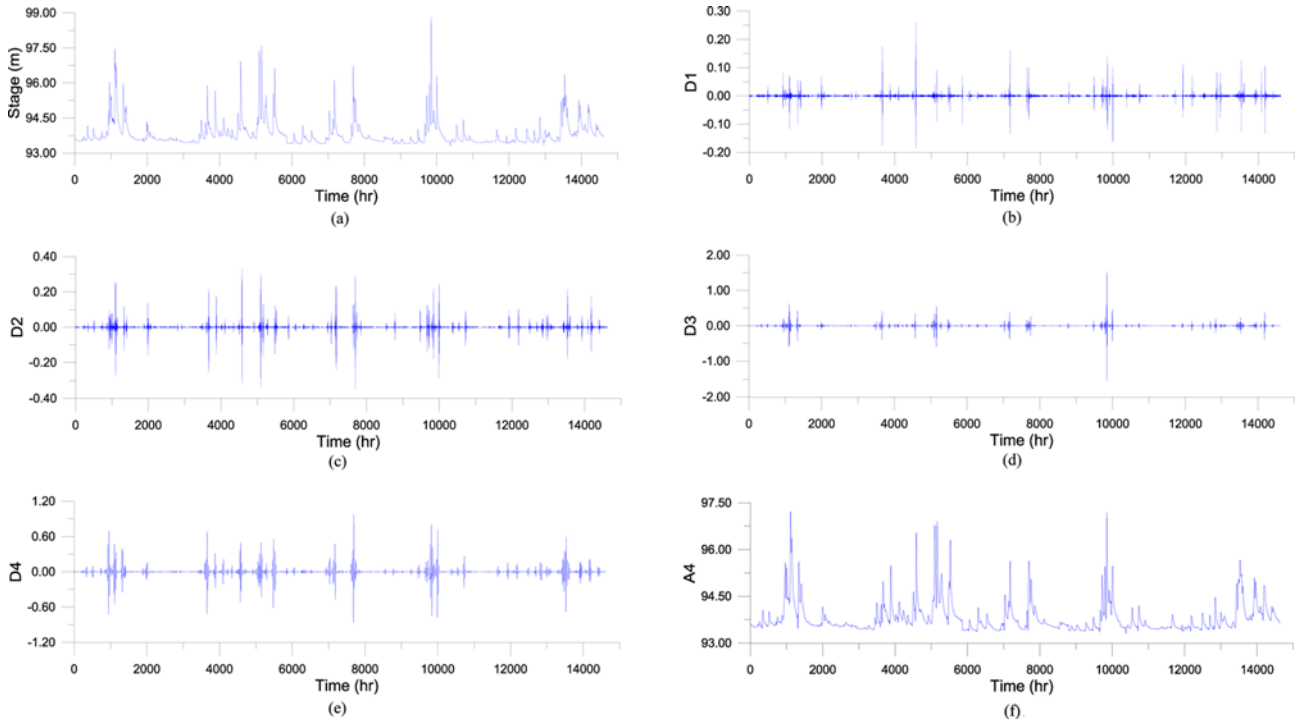

Fig. 7. Original and Decomposed Time Series ($D1$, $D2$, $D3$ and $A3$) using *db*10 Mother Wavelet: (a) Original time series, (b) $D1$, (c) $D2$, (d) $D3$, (e) $D4$, (f) $A4$

be written as follows:

$$Q_{t+i} = f(D1_{t-8}, D2_{t-8}, D3_{t-8}, D4_{t-8}, A4_{t-8}, D1_{t-7}, D2_{t-7}, D3_{t-7}, D4_{t-7}, A4_{t-7}, D1_{t-6}, D2_{t-6}, D3_{t-6}, D4_{t-6}, A4_{t-6}, D1_{t-5}, D2_{t-5}, D3_{t-5}, D4_{t-5}, A4_{t-5}, D1_{t-4}, D2_{t-4}, D3_{t-4}, D4_{t-4}, A4_{t-4}, D1_{t-3}, D2_{t-3}, D3_{t-3}, D4_{t-3}, A4_{t-3}, D1_{t-2}, D2_{t-2}, D3_{t-2}, D4_{t-2}, A4_{t-2}, D1_{t-1}, D2_{t-1}, D3_{t-1}, D4_{t-1}, A4_{t-1}, D1_t, D2_t, D3_t, D4_t, A4_t) \quad (19)$$

where $D1_{t-j}$, $D2_{t-j}$, $D3_{t-j}$, $D4_{t-j}$, $A4_{t-j}$ are input variables ($j = 0, 1, \ldots, 8$) and Q_{t+i} are output variables for lead time i hour ($i = 1, 2, \ldots, 10$).

For forecasting flood stage values using ANN and WANN models, the MLP neural network model was used. The number of nodes in the hidden layer was determined using a trial-and-error approach from the previous studies of MLP (Kisi, 2007; Kim *et al*., 2012; Kim *et al.*, 2013). The optimal size of hidden nodes was determined by investigating the RMSE values for different number of hidden nodes from 1 to 20. Fig. 8 shows the RMSE values for different numbers of hidden nodes. Three hidden nodes were selected as the optimal size of hidden nodes with minimum RMSE value. In this study, the ANN and WANN models were trained using the vanilla backpropagation algorithm (Bergmeir and Benítez, 2012). The logistic sigmoid activation function was selected for computing the output of each neuron. According to the algorithm chosen, the data of input and output nodes were scaled to the range of [0, 1].

For applying the ANFIS and WANFIS models, the FIS of Sugeno type was used in this study. The ANFIS model has the drawback that the number of control rules increases rapidly and running time also increases exponentially as the number of input variables and MFs increases. To overcome the drawback, the FIS structure was generated using the subtractive clustering algorithm (MathWorks, 2014). The generated FIS was trained using a hybrid learning algorithm combining the least-squares and backpropagation gradient descent methods. The ability of the ANFIS model to achieve the performance goal depends on predefined internal parameters, including the number and shape of MFs and the step size (El-Shafie *et al*., 2007). In this study, we selected three Gaussian MFs for each input node and a linear function as output MFs using a trial-and-error approach. Since the different values of step size did not largely affect the performance of models, we used default values, initial step size of 0.01, step size decrease rate of 0.9 and step size increase rate of 1.1, suggested by MathWorks (2014).

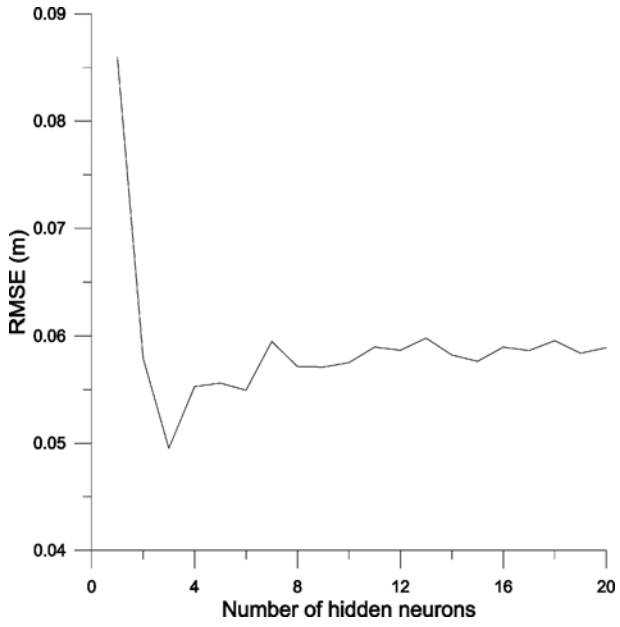

Fig. 8. RMSE Estimates for Different Numbers of Hidden Neurons

4.2 Performance Evaluation

In this study, the performance of models applied for flood forecasting was evaluated using seven performance indexes, CE, d, r^2, RMSE, MAE, MSE and MSRE. Larger values of CE, d and r^2 and smaller values of RMSE, MAE, MSE and MSRE indicate that the efficiency of a model is higher than that of other models.

Figure 9 shows the performance of ANN, ANFIS, WANN and WANFIS models for different lead times from 1 to 10 hours. Tables 1-4 summarize the values of performance indexes for the models. The ANN and ANFIS models yield almost similar results in terms of performance indexes except for MAE. The values of MAE of the ANFIS model were slightly lower than those of the ANN model for all lead times. These indicate that the performances of the ANN and ANFIS models were similar in terms of performance indexes.

The values of CE, r^2 and d of the WANN and WANFIS models were higher than those of the ANN and ANFIS models. The values of RMSE, MAE, MSE and MSRE of the WANN and WANFIS models were lower than those of the ANN and ANFIS models. These indicated that the WANN and WANFIS models were superior to the ANN and ANFIS models in terms of model efficiency. The WANN and WANFIS models used sub-time series, decomposed by DWT, as input data of the models, while the ANN and ANFIS models used original flood time series as input data of the models without decomposition. These indicated that the WANN and WANFIS models using sub-time series decomposed by DWT can yield higher model efficiency than the ANN and ANFIS models using original flood input data without wavelet decomposition. These also indicated that using sub-time series decomposed by DWT as input data of the ANN and ANFIS models can improve the performance of the models.

The performance of ANN, ANFIS, WANN and WANFIS models was evaluated for different lead times of 1 to 10 hours. From Fig. 9, the slopes of curves for CE, d and r^2 increased negatively, while the slopes of curves for RMSE, MAE, MSE and MSRE increased positively, as the lead time increased. For CE, d and r^2, the slopes of curves for the WANN and WANFIS models gradually increased negatively, while the slopes of curves for the ANN and ANFIS models drastically increased negatively, as the lead time increased. For RMSE and MAE, the slopes of curves for the ANN and ANFIS models were greater

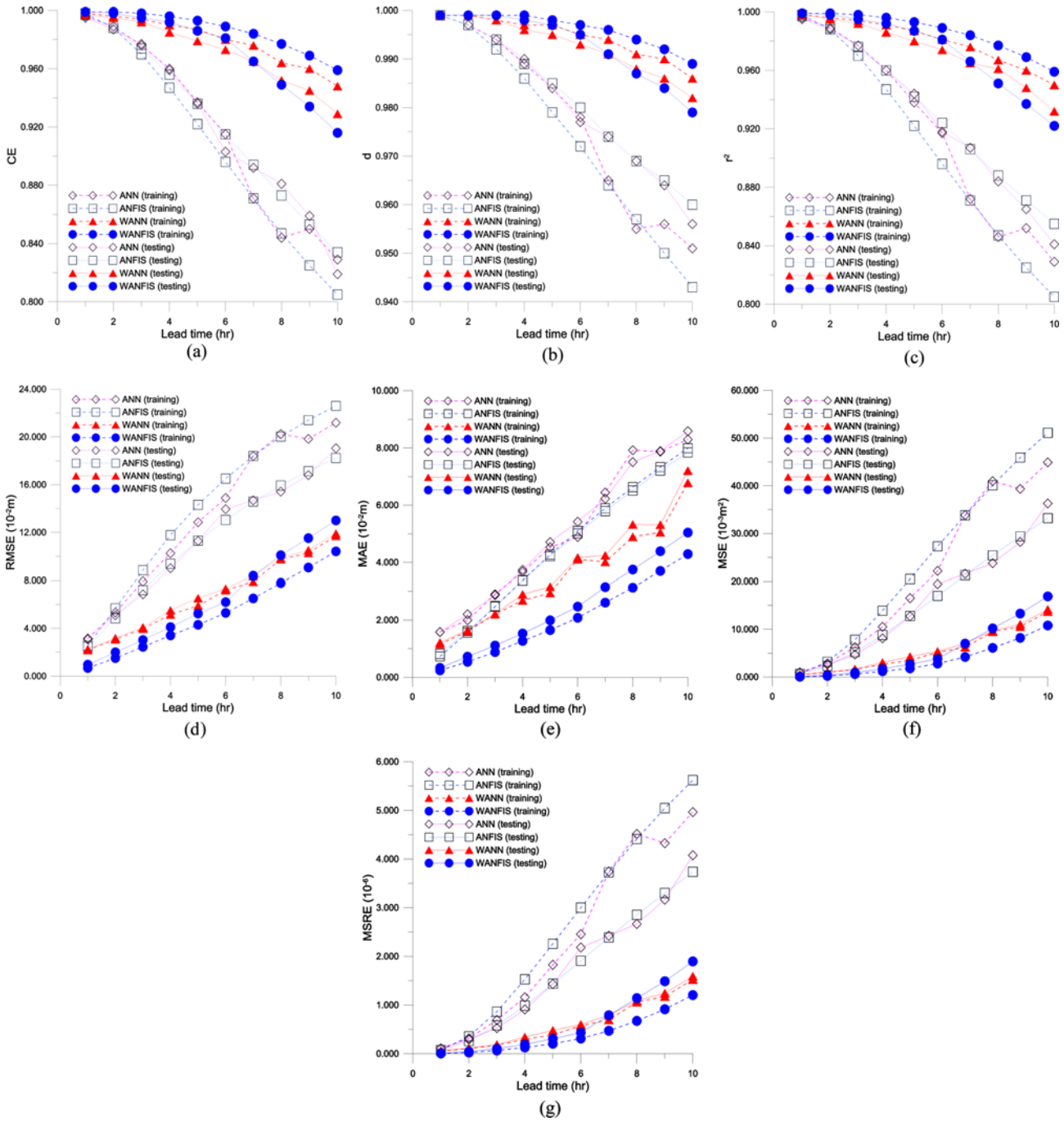

Fig. 9. Comparison of Model Performance for Different Lead Times: (a) CE, (b) *d*, (c) r^2, (d) RMSE, (e) MAE, (f) MSE, (g) MSRE

than those for the WANN and WANFIS models. For MSE and MSRE, the slopes of curves for the ANN and ANFIS models drastically increased positively, while the slopes of curves for the WANN and WANFIS models gradually increased positively, as the lead time increased. The differences between the values of performance indexes for the WANN and WANFIS models and those for the ANN and ANFIS models significantly increased as the lead time increased. These indicate that using sub-time series decomposed by DWT as input data of the ANN and ANFIS models can improve the performance of the models for different lead times, and especially for higher lead times.

For lead times of 1 to 6 hours, the values of CE, *d* and r^2 for the WANFIS model were slightly greater than those for the WANN model, and the values of RMSE, MSE and MSRE for the WANFIS model were slightly lower than those for the WANN model. For lead time of 7 hours, the values of performance indexes for the WANN and WANFIS models were similar. For lead times of 8 to 10 hours, the values of CE, *d* and r^2 for the WANFIS model were slightly lower than those for the WANN model and the values of RMSE, MSE and MSRE for the

Table 1. Performance of the ANN Model for Different Lead Times

Lead Time (hr)		CE	d	r^2	RMSE (10^{-2} m)	MAE (10^{-2} m)	MSE (10^{-3} m^2)	MSRE (10^{-6})
Training	1	0.996	0.999	0.996	3.162	1.585	1.000	0.111
	2	0.989	0.997	0.989	5.333	1.994	2.844	0.315
	3	0.976	0.994	0.977	7.921	2.901	6.274	0.692
	4	0.960	0.989	0.960	10.259	3.673	10.524	1.160
	5	0.937	0.984	0.938	12.853	4.524	16.521	1.826
	6	0.915	0.978	0.917	14.885	4.884	22.155	2.455
	7	0.871	0.965	0.872	18.397	6.442	33.845	3.744
	8	0.844	0.955	0.846	20.239	7.920	40.963	4.515
	9	0.850	0.956	0.852	19.835	7.870	39.341	4.326
	10	0.829	0.951	0.829	21.189	8.297	44.896	4.964
Testing	1	0.995	0.999	0.995	3.087	1.579	0.953	0.107
	2	0.987	0.997	0.988	5.100	2.203	2.601	0.293
	3	0.977	0.994	0.977	6.854	2.871	4.698	0.527
	4	0.959	0.990	0.960	9.004	3.738	8.107	0.910
	5	0.936	0.984	0.944	11.284	4.708	12.734	1.431
	6	0.903	0.977	0.918	13.950	5.420	19.459	2.183
	7	0.892	0.974	0.907	14.684	6.225	21.563	2.423
	8	0.881	0.969	0.884	15.415	7.505	23.762	2.665
	9	0.859	0.964	0.865	16.795	7.889	28.209	3.166
	10	0.819	0.956	0.841	19.050	8.587	36.289	4.077

Table 2. Performance of the ANFIS Model for Different Lead Times

Lead Time (hr)		CE	d	r^2	RMSE (10^{-2} m)	MAE (10^{-2} m)	MSE (10^{-3} m^2)	MSRE (10^{-6})
Training	1	0.998	0.999	0.998	2.560	0.730	0.654	0.072
	2	0.988	0.997	0.988	5.680	1.560	3.200	0.356
	3	0.970	0.992	0.970	8.840	2.470	7.800	0.863
	4	0.947	0.986	0.947	11.770	3.380	13.900	1.528
	5	0.922	0.979	0.922	14.310	4.270	20.500	2.259
	6	0.896	0.972	0.896	16.510	5.100	27.300	3.002
	7	0.871	0.964	0.871	18.400	5.890	33.800	3.727
	8	0.847	0.957	0.847	20.030	6.630	40.100	4.414
	9	0.825	0.950	0.825	21.410	7.330	45.900	5.045
	10	0.805	0.943	0.805	22.600	7.990	51.100	5.622
Testing	1	0.997	0.999	0.997	2.460	0.820	0.605	0.068
	2	0.988	0.997	0.989	4.840	1.620	2.300	0.263
	3	0.974	0.994	0.976	7.200	2.490	5.200	0.582
	4	0.956	0.989	0.960	9.380	3.370	8.800	0.987
	5	0.936	0.985	0.942	11.310	4.210	12.800	1.438
	6	0.915	0.980	0.924	13.040	5.010	17.000	1.911
	7	0.894	0.974	0.906	14.580	5.790	21.300	2.390
	8	0.873	0.969	0.888	15.930	6.510	25.400	2.853
	9	0.853	0.965	0.871	17.130	7.210	29.400	3.301
	10	0.834	0.960	0.855	18.230	7.840	33.200	3.738

WANFIS model were slightly greater than those for the WANN model. These indicate that the WANFIS model yielded slightly better prediction compared with the WANN model for lead times of 1 to 6 hours, while the WANN model yielded slightly better predictions compared with the WANFIS model for lead times of 8 to 10 hours.

Figures 10-12 show scatter plots of the ANN, ANFIS, WANN and WANFIS models for lead times, 1, 3 and 5 hours. Figures 13-15 show stage hydrographs of the models. Standard deviations around $y = x$ line (blue line) for the WANN model were lower than those for the ANN model, and standard deviations around $y = x$ line (blue line) for the WANFIS model

Table 3. Performance of the WANN Model for Different Lead Times

Lead Time (hr)		CE	d	r^2	RMSE (10^{-2} m)	MAE (10^{-2} m)	MSE (10^{-3} m^2)	MSRE (10^{-6})
Training	1	0.998	0.999	0.998	2.187	1.152	0.478	0.053
	2	0.996	0.999	0.996	3.083	1.606	0.950	0.106
	3	0.994	0.998	0.994	3.977	2.205	1.582	0.175
	4	0.990	0.997	0.991	5.160	2.684	2.663	0.296
	5	0.987	0.997	0.987	5.897	2.942	3.478	0.387
	6	0.980	0.995	0.981	7.201	4.107	5.185	0.576
	7	0.976	0.994	0.976	7.890	4.014	6.225	0.694
	8	0.964	0.991	0.967	9.745	4.891	9.497	1.060
	9	0.960	0.990	0.960	10.276	5.051	10.560	1.175
	10	0.948	0.986	0.950	11.708	6.781	13.708	1.525
Testing	1	0.997	0.999	0.998	2.263	1.216	0.512	0.058
	2	0.995	0.999	0.995	3.189	1.641	1.017	0.115
	3	0.992	0.998	0.992	4.093	2.217	1.675	0.188
	4	0.985	0.996	0.986	5.512	2.897	3.038	0.342
	5	0.979	0.995	0.980	6.532	3.165	4.267	0.479
	6	0.973	0.993	0.974	7.326	4.170	5.367	0.601
	7	0.965	0.991	0.965	8.385	4.243	7.031	0.789
	8	0.952	0.988	0.961	9.842	5.324	9.686	1.090
	9	0.945	0.986	0.948	10.521	5.314	11.070	1.243
	10	0.929	0.982	0.932	11.905	7.205	14.173	1.592

Table 4. Performance of the WANFIS Model for Different Lead Times

Lead Time (hr)		CE	d	r^2	RMSE (10^{-2} m)	MAE (10^{-2} m)	MSE (10^{-3} m^2)	MSRE (10^{-6})
Training	1	0.999	0.999	0.999	0.680	0.240	0.046	0.005
	2	0.999	0.999	0.999	1.510	0.540	0.227	0.025
	3	0.998	0.999	0.998	2.450	0.880	0.602	0.067
	4	0.996	0.999	0.996	3.410	1.270	1.200	0.128
	5	0.993	0.998	0.993	4.300	1.650	1.800	0.205
	6	0.989	0.997	0.989	5.300	2.080	2.800	0.312
	7	0.984	0.996	0.984	6.510	2.610	4.200	0.471
	8	0.977	0.994	0.977	7.790	3.130	6.100	0.674
	9	0.969	0.992	0.969	9.060	3.700	8.200	0.913
	10	0.959	0.989	0.959	10.410	4.290	10.800	1.205
Testing	1	0.999	0.999	0.999	0.960	0.330	0.093	0.010
	2	0.998	0.999	0.998	2.000	0.720	0.399	0.045
	3	0.995	0.999	0.995	3.030	1.110	0.917	0.103
	4	0.992	0.998	0.992	4.110	1.530	1.700	0.191
	5	0.986	0.997	0.987	5.240	1.990	2.700	0.309
	6	0.981	0.995	0.981	6.200	2.470	3.800	0.432
	7	0.965	0.991	0.966	8.380	3.150	7.000	0.789
	8	0.949	0.987	0.951	10.090	3.750	10.200	1.141
	9	0.934	0.984	0.937	11.520	4.390	13.300	1.488
	10	0.916	0.979	0.922	13.000	5.040	16.900	1.896

were lower than those for the ANFIS model, for different lead times. When $y = ax + b$ lines (red lines) fitted to the scatter graphs were examined for the models, 'a' for the WANFIS model get closer to the value 1, and 'b' for the WANFIS model get closer the value 0, for different lead times. These indicated that the WANFIS and WANN models yielded better prediction compared with the ANN and ANFIS models for different lead times. These also indicated that wavelet decomposition can improve the efficiency of the ANN and ANFIS models.

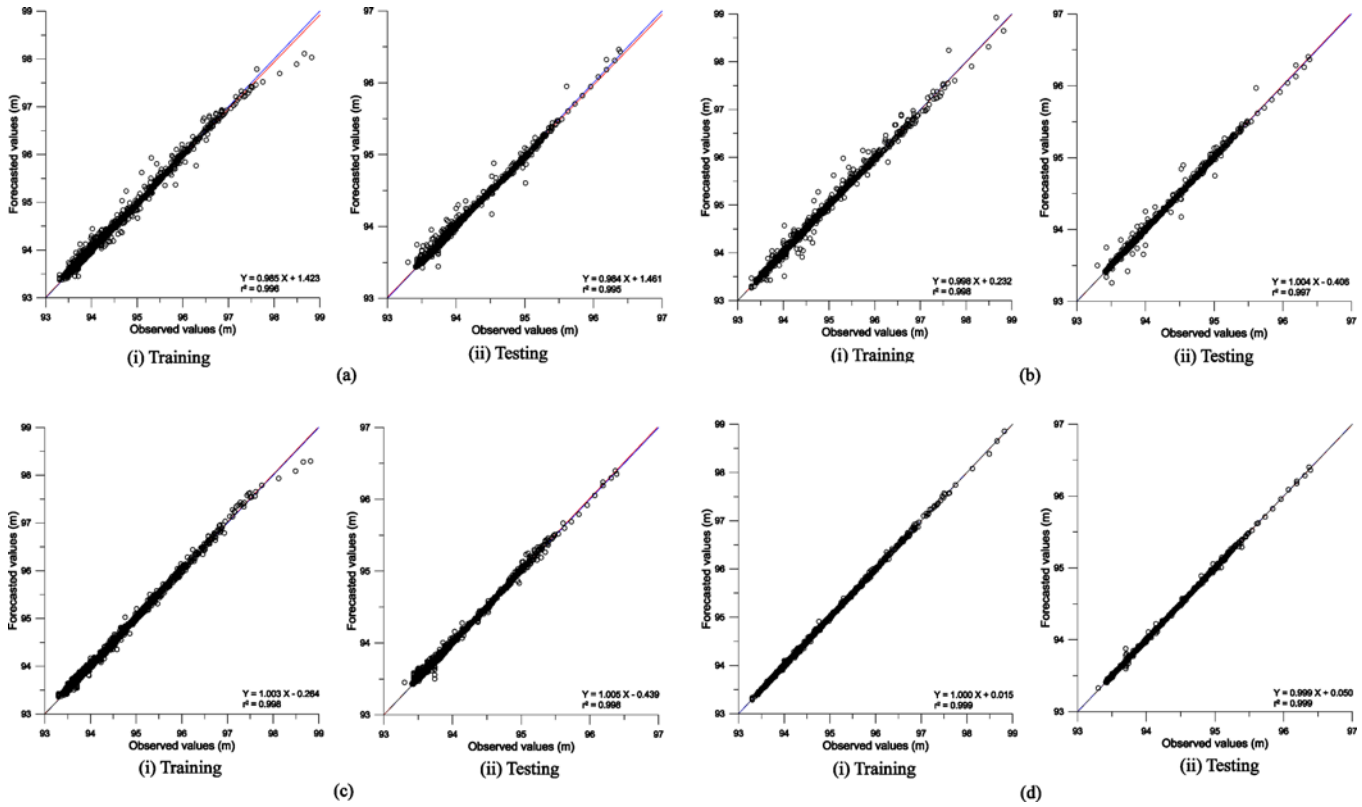

Fig. 10. Scatter Plots (lead time = 1 hr): (a) ANN, (b) ANFIS, (C) WANN, (d) WANFIS

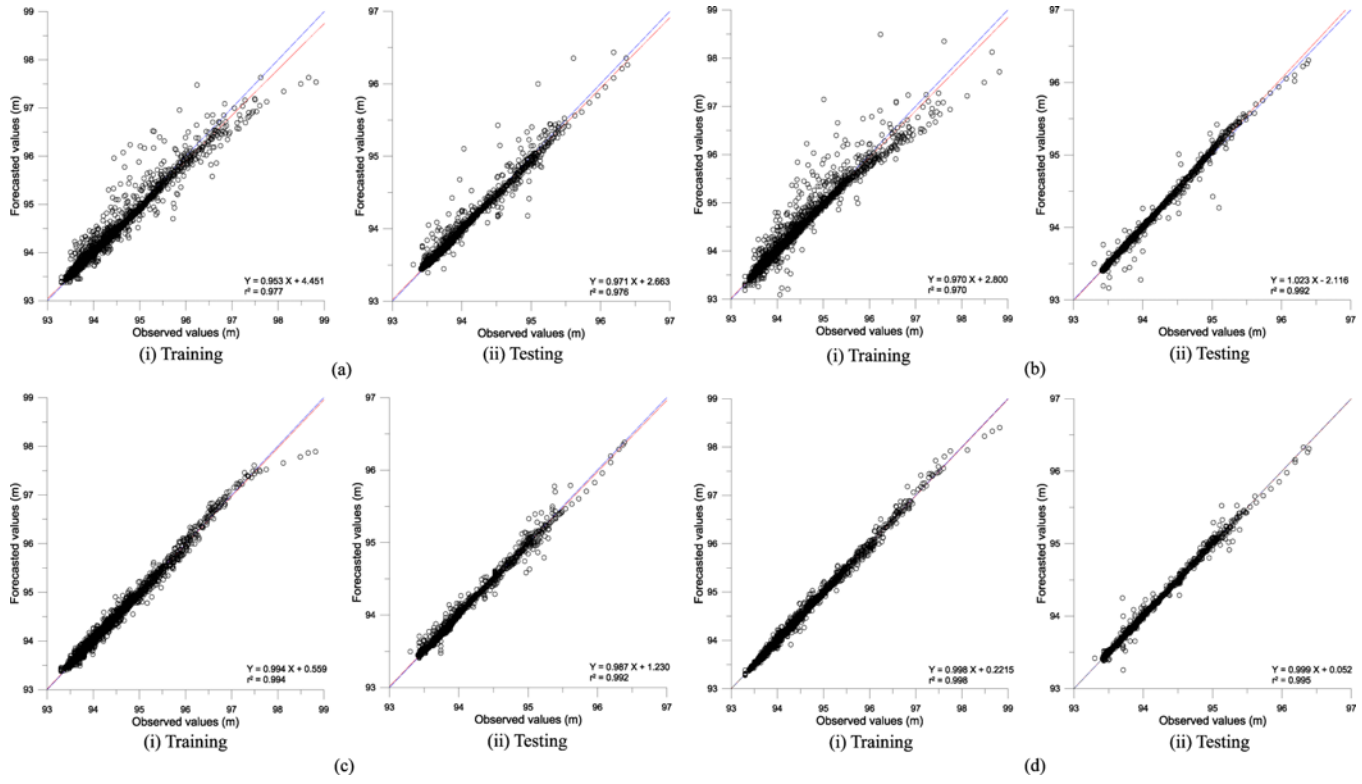

Fig. 11. Scatter Plots (lead time = 3 hr): (a) ANN, (b) ANFIS, (c) WANN, (d) WANFIS

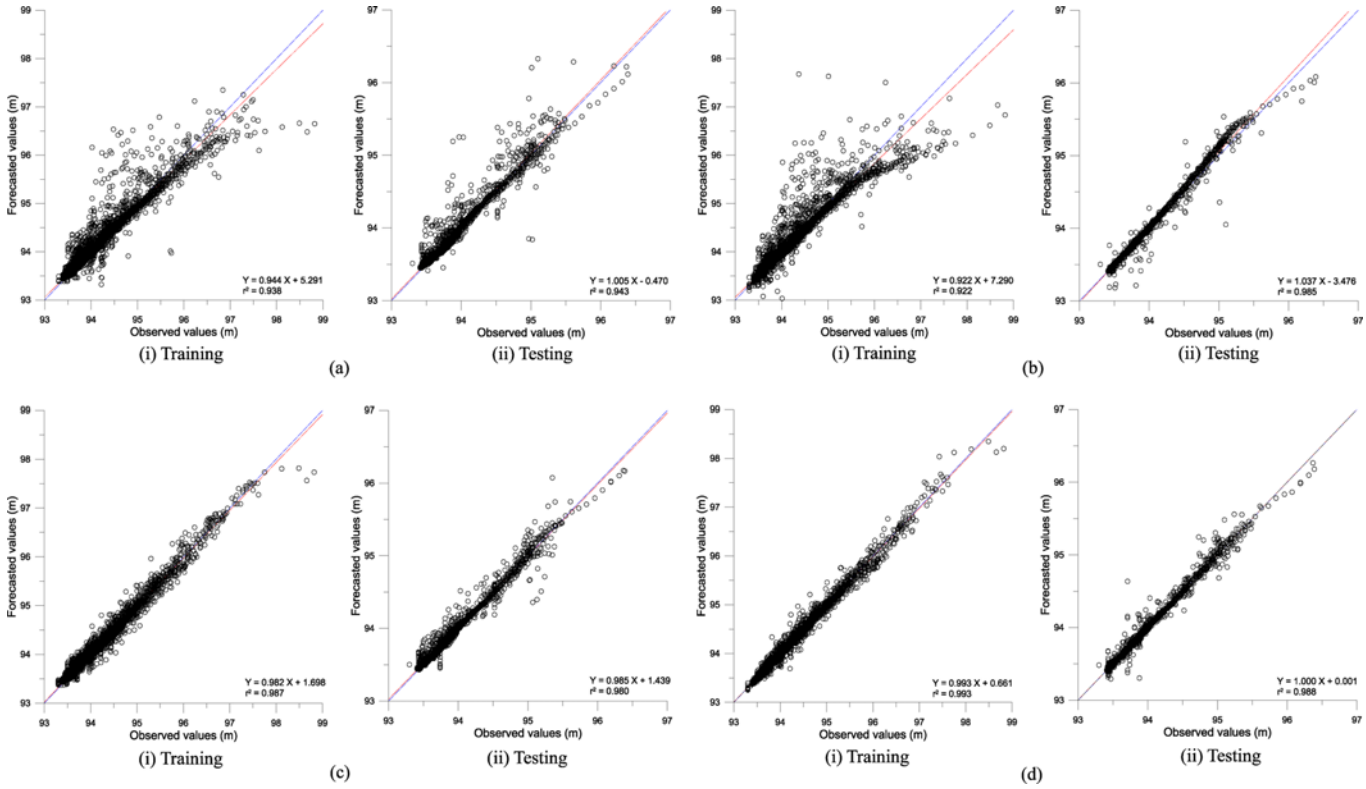

Fig. 12. Scatter Plots (lead time = 5 hr): (a) ANN, (b) ANFIS, (c) WANN, (d) WANFIS

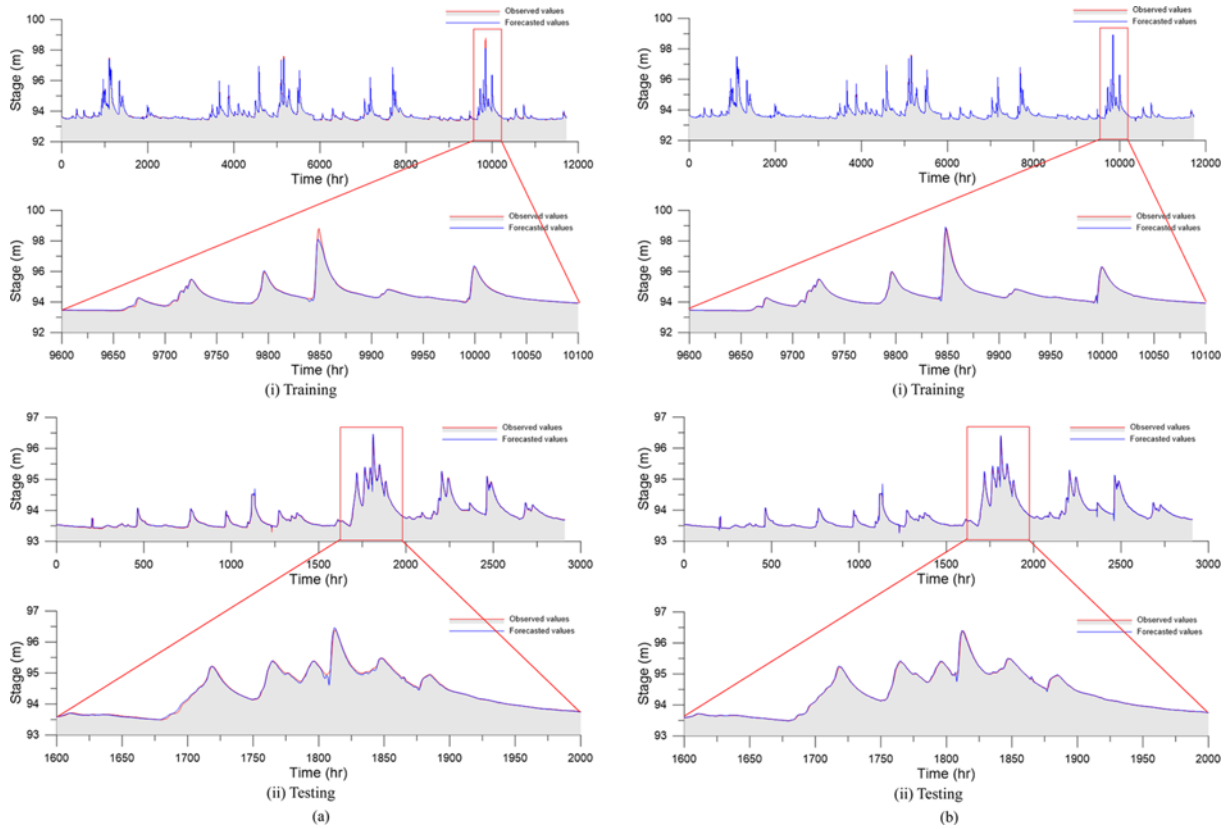

Fig. 13. Stage Hydrographs (lead time = 1 hr): (a) ANN, (b) ANFIS

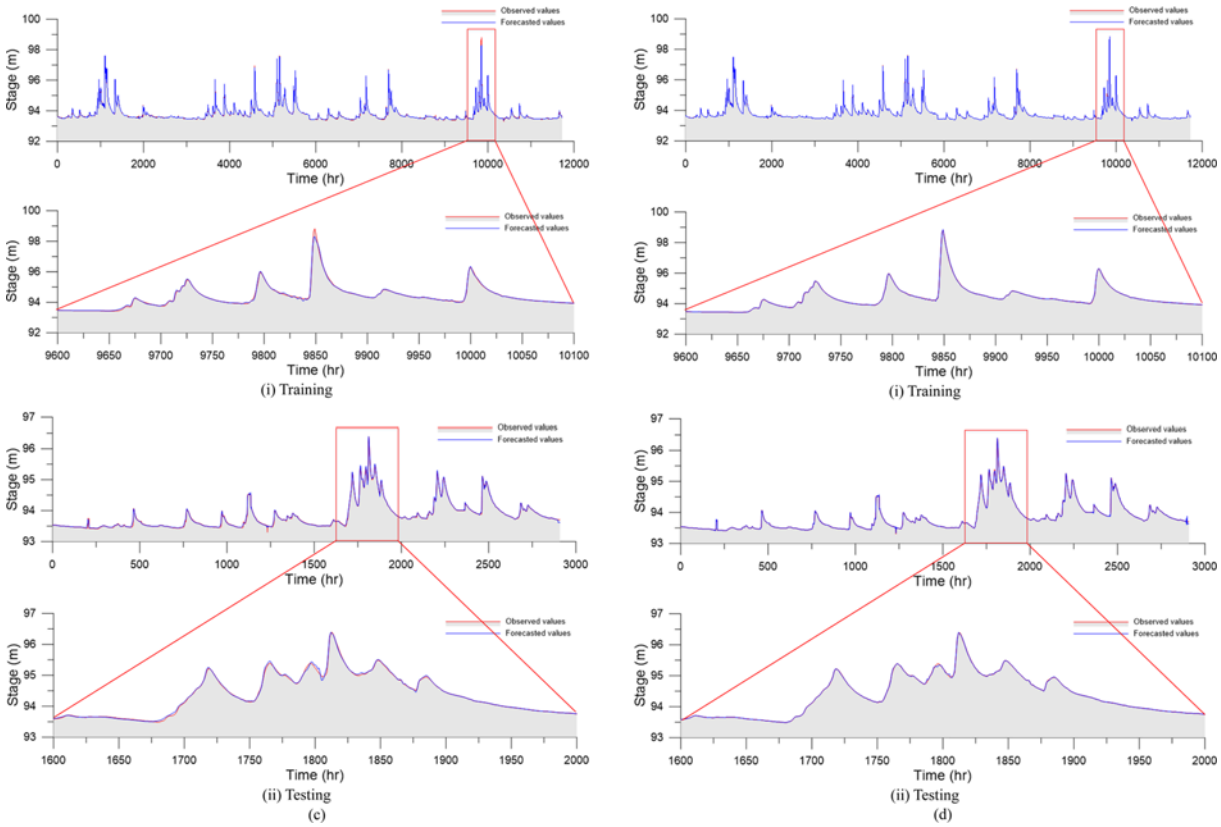

Fig. 13. continued: (c) WANN, (d) WANFIS

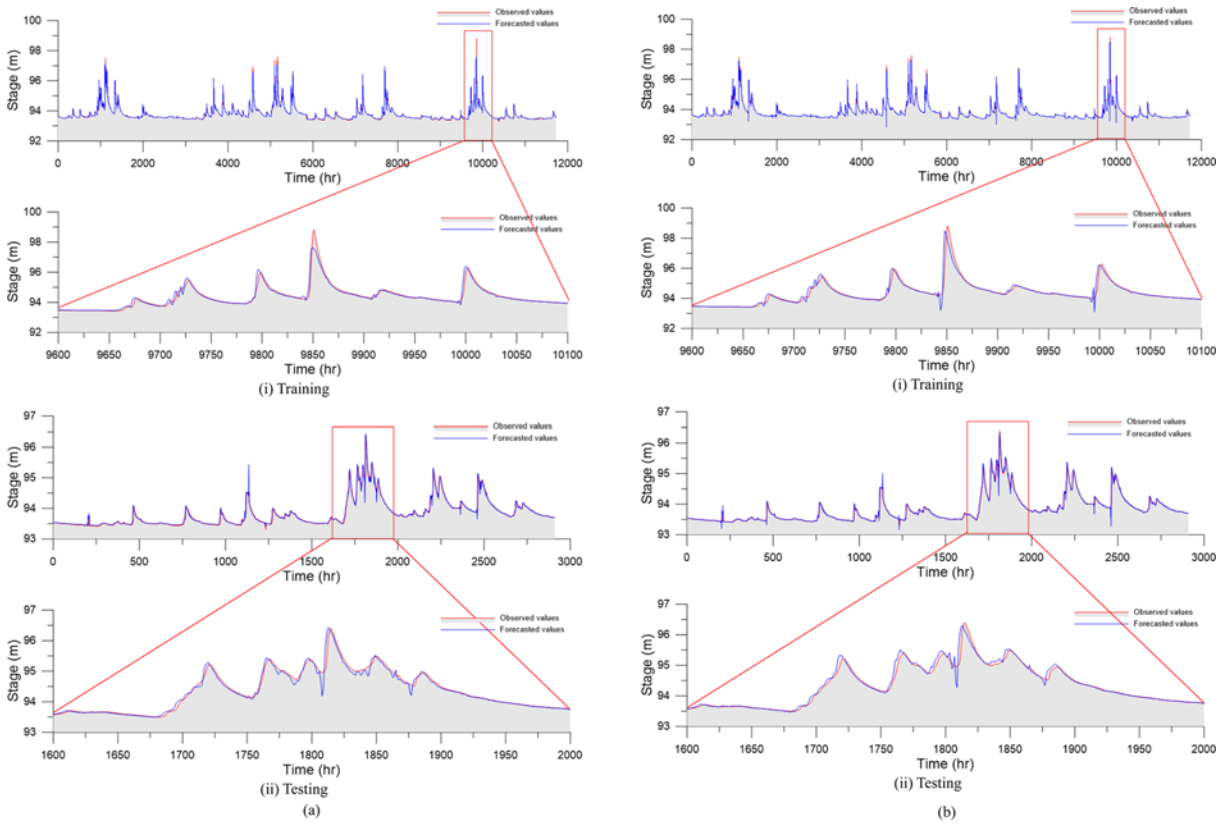

Fig. 14. Stage Hydrographs (lead time = 3 hr): (a) ANN, (b) ANFIS

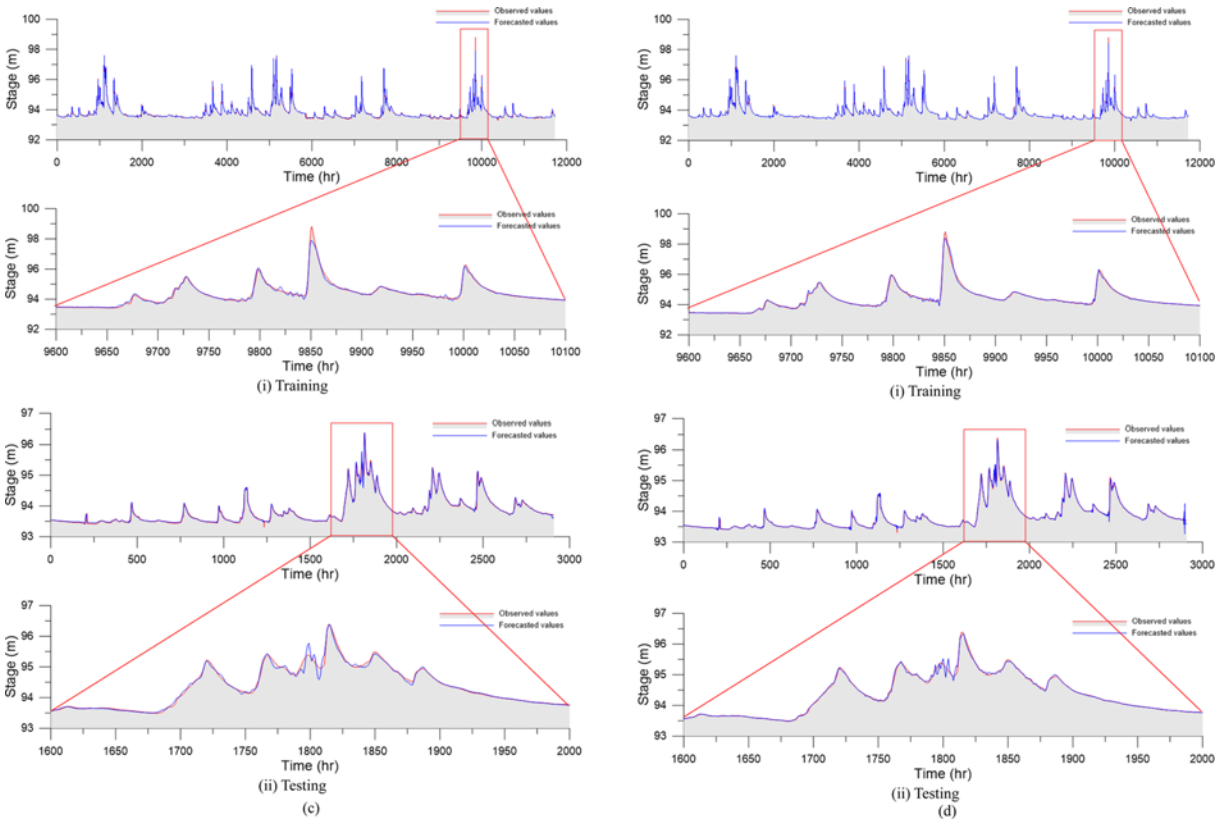

Fig. 14. continued: (a) WANN, (d) WANFIS

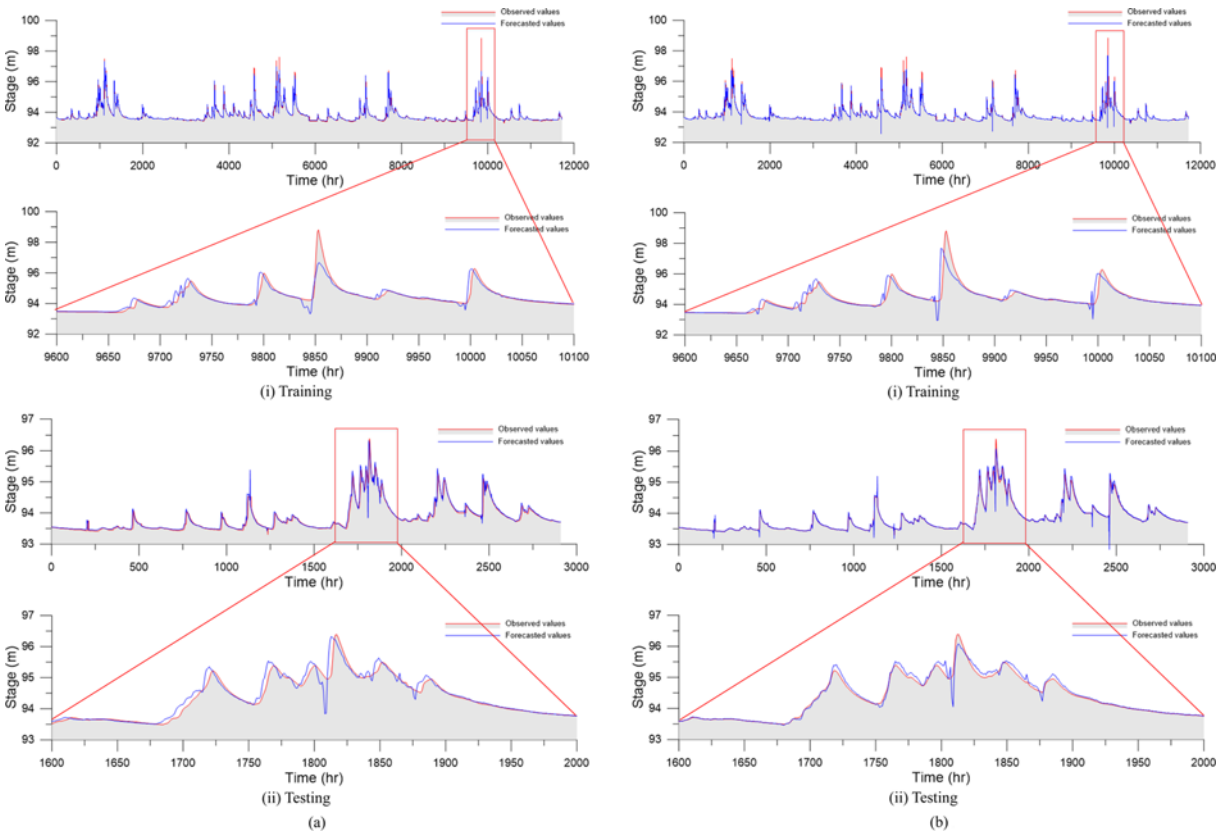

Fig. 15. Stage Hydrographs (lead time = 5 hr): (a) ANN, (b) ANFIS

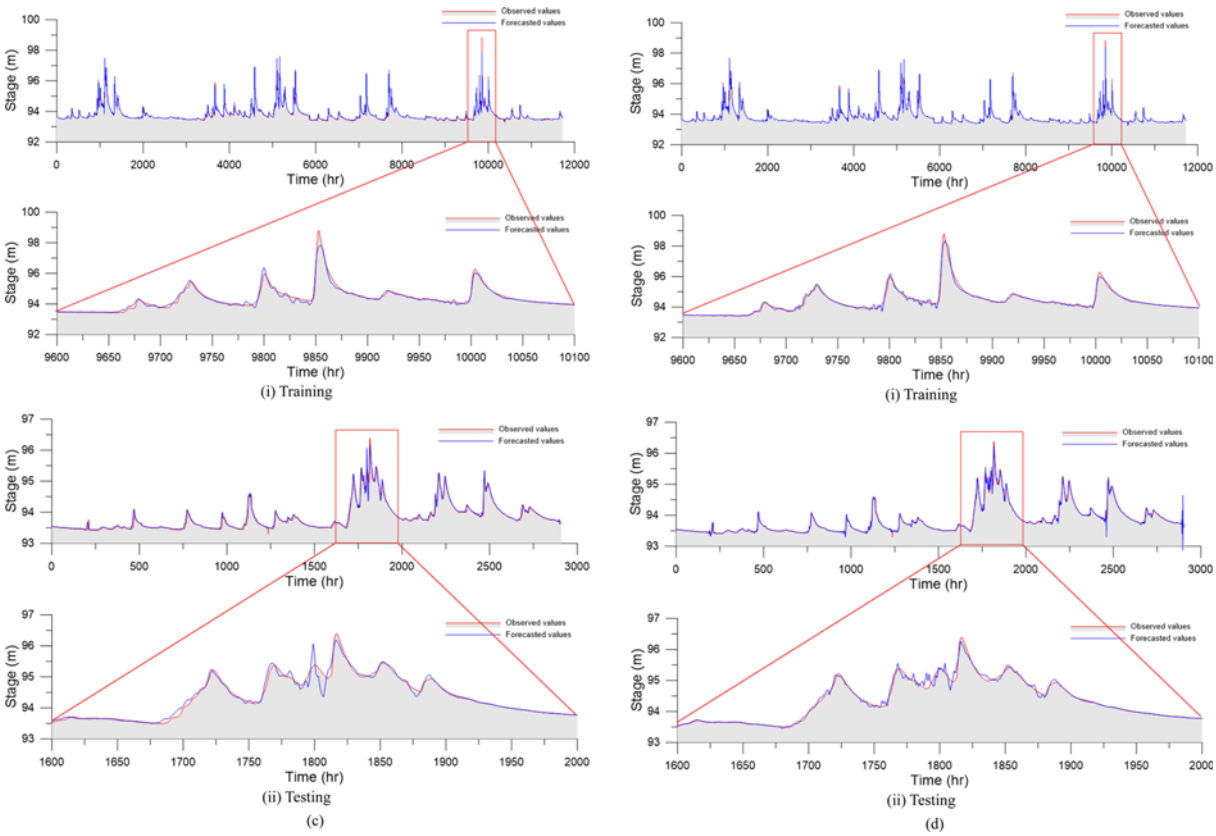

Fig. 15. continued: (c) WANN, (d) WANFIS

5. Conclusions

This study applies two hybrid models, the Wavelet-based Artificial Neural Network (WANN) and the Wavelet-Based Adaptive Neuro-Fuzzy Inference System (WANFIS), the flood forecasting in the Bocheong stream catchment, South Korea. It then evaluates the forecast accuracy of these models for different lead times, based on seven performance indexes, including the Coefficient of Efficiency (CE), the index of agreement (d), the coefficient of determination (r^2), the Root-Mean-Square Error (RMSE), the Mean Absolute Error (MAE), the Mean Squared Error (MSE) and the Mean Squared Relative Error (MSRE).

The WANN and WANFIS models yield better results than do the ANN and ANFIS models for different lead times. In terms of model efficiency, the WANFIS model is found to be superior to other models for lead times of 1 to 6 hours, and the WANN model is found to be superior to other models for lead times of 8 to 10 hours. The WANN and WANFIS models yield similar results and are superior to other models for lead time of 7 hours. Results show that the combination of wavelet decomposition and data-driven models, including the ANN and ANFIS, can improve the efficiency of data-driven models. Results also indicate that the combination of wavelet decomposition and data-driven models can be a potential tool for accurate forecasting flood stage.

For further studies, the WANN and WANFIS models can be applied to forecasting hydrological variables of other watersheds under different climate, geographical and hydrological conditions. It is suggested to develop hybrid models combining wavelet decomposition with other data-driven models and evolutionary algorithms for forecasting hydrological variables with non-stationary and non-linear relationships.

References

Adamowski, J. (2008). "Development of a short-term river flood forecasting method for snowmelt driven floods based on wavelet and cross-wavelet analysis." *Journal of Hydrology*, Vol. 353, Issues 3-4, pp. 247-266, DOI: 10.1016/j.jhydrol.2008.02.013.

Adamowski, J. and Chan, H. F. (2011). "A wavelet neural network conjunction model for groundwater level forecasting." *Journal of Hydrology*, Vol. 407, Issues 1-4, pp. 28-40, DOI: 10.1016/j.jhydrol.2011.06.013.

Adamowski, J. and Sun, K. (2010). "Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds." *Journal of Hydrology*, Vol. 390, Issues 1-2, pp. 85-91, DOI: 10.1016/j.jhydrol.2010.06.033.

Baratti, R., Cannas, B., Fanni, A., Pintus, M., Sechi, G. M., and Toreno, N. (2003). "River flow forecast for reservoir management through neural networks." *Neurocomputing*, Vol. 55, Issues 3-4, pp. 421-437, DOI: 10.1016/S0925-2312(03)00387-4.

Belayneh, A. and Adamowski, J. (2012). "Standard precipitation index drought forecasting using neural networks, wavelet neural networks, and support vector regression." *Applied Computational Intelligence*

and Soft Computing, Vol. 2012, Article ID 794061, DOI: 10.1155/2012/794061.

Bergmeir, C. and Benítez, J. M. (2012). "Neural networks in R using the Stuttgart neural network simulator: RSNNS." *Journal of Statistical Software*, Vol. 46, Issue 7, pp. 1-26.

Brath, A. and Rosso, R. (1993). "Adaptive calibration of a conceptual model for flash flood forecasting." *Water Resources Research*, Vol. 29, Issue 8, pp. 2561-2572, DOI: 10.1029/93WR00665.

Calvo, B. and Savi, F. (2009). "Real-time flood forecasting of the Tiber river in Rome." *Natural Hazards*, Vol. 50, Issue 3, pp. 461-477, DOI: 10.1007/s11069-008-9312-9.

Campolo, M., Soldati, A., and Andreussi, P. (2003). "Artificial neural network approach to flood forecasting in the River Arno." *Hydrological Sciences Journal*, Vol. 48, Issue 3, pp. 381-398, DOI: 10.1623/hysj.48.3.381.45286.

Catalão, J. P. S., Pousinho, H. M. I., and Mendes, V. M. F. (2011). "Hybrid wavelet-PSO-ANFIS approach for short-term electricity prices forecasting." *IEEE Transactions on Power Systems*, Vol. 26, Issue 1, pp. 137-144, DOI: 10.1109/TPWRS.2010.2049385.

Chang, F. J., Chiang, Y. M., and Chang, L. C. (2007). "Multi-step-ahead neural networks for flood forecasting." *Hydrological Sciences*, Vol. 52, Issue 1, pp. 114-130, DOI: 10.1623/hysj.52.1.114.

Chang, F. J. and Hwang, Y. Y. (1999). "A self-organization algorithm for real-time flood forecast." *Hydrological Processes*, Vol. 13, Issue 2, pp. 123-138, DOI: 10.1002/(SICI)1099-1085(19990215)13:2<123::AID-HYP701>3.0.CO;2-2.

Chiang, Y. M., Hsu, K. L., Chang, F. J., Hong, Y., and Sorooshian, S. (2007). "Merging multiple precipitation sources for flash flood forecasting." *Journal of Hydrology*, Vol. 340, Issues 3-4, pp. 183-196, DOI: 10.1016/j.jhydrol.2007.04.007.

Dadu, K. S. and Deka, P. C. (2013). "Multistep lead time forecasting of hydrologic time series using Daubechies wavelet – Neural network hybrid model." *International Journal of Scientific and Engineering Research*, Vol. 4, Issue 10, pp. 115-124.

Dawson, C. W. and Wilby, R. L. (2001). "Hydrological modelling using artificial neural networks." *Progress in Physical Geography*, Vol. 25, No. 1, pp. 80-108. DOI: 10.1177/030913330102500104.

Deshmukh, R. P., and Ghatol, A. A. (2010). "Short term flood forecasting using general recurrent neural network modeling a comparative study." *International Journal of Computer Applications*, Vol. 8, No. 12, pp. 5-9, DOI: 10.5120/1259-1777.

Duan, Q., Sorooshian, S., and Gupta, V. K. (1992). "Effective and efficient global optimization for conceptual rainfall-runoff models." *Water Resources Research*, Vol. 28, Issue 4, pp. 1015-1031, DOI: 10.1029/91WR02985.

El-Shafie, A., Taha, M. R., and Noureldin, A. (2007). "A neuro-fuzzy model for inflow forecasting of the Nile river at Aswan high dam." *Water Resources Management*, Vol. 21, Issue 3, pp. 533-556, DOI: 10.1007/s11269-006-9027-1.

Grayson, R. B., Moore, I. D., and McMahon, T. A. (1992). "Physically based hydrologic modeling: 2. Is the concept realistic?." *Water Resources Research*, Vol. 28, Issue 10, pp. 2659-2666, DOI: 10.1029/92WR01259.

Günther, F., and Fritsch, S. (2010). "Neuralnet: Training of neural networks." *The R Journal*, Vol. 2, Issue 1, pp. 30-38.

Hornik, K., Stichcombe, M., and White, H. (1989). "Multilayer feedforward networks are universal approximators." *Neural Networks*, Vol. 2, Issue 5, pp. 359-366, DOI: 10.1016/0893-6080(89)90020-8.

Imrie, C. E., Durucan, S., and Korre, A. (2000). "River flow prediction using artificial neural networks: Generalisation beyond the calibration range." *Journal of Hydrology*, Vol. 233, Issues 1-4, pp. 138-153, DOI: 10.1016/S0022-1694(00)00228-6.

Jang, J. S. R. (1993). "ANFIS: Adaptive-network-based fuzzy inference system." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, Issue 3, pp. 665-685, DOI : 10.1109/21.256541.

Jang, J. S. R., Sun, C. T., and Mizutani, E. (1997*). Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence*, Prentice-Hall, New Jersey.

Kashani, M. H., Montaseri, M., and Yaghin, M. A. L. (2007). "Flood estimation at ungauged sites using a new nonlinear regression model and artificial neural networks." *American-Eurasian Journal of Agricultural and Environmental Sciences*, Vol. 2, Issue 6, pp. 784-791.

Kim, S., and Kim, H. S. (2008). "Uncertainty reduction of the flood stage forecasting using neural networks model." *Journal of the American Water Resources Association*, Vol. 44, Issue 1, pp. 148-165, DOI: 10.1111/j.1752-1688.2007.00144.x.

Kim, S., Shiri, J., and Kisi, O. (2012). "Pan evaporation modeling using neural computing approach for different climatic zones." *Water Resources Management*, Vol. 26, Issue 11, pp. 3231-3249, DOI: 10.1007/s11269-012-0069-2.

Kim, S., Shiri, J., Kisi, O., and Singh, V. P. (2013). "Estimating daily pan evaporation using different data-driven methods and lag-time patterns." *Water Resources Management*, Vol. 27, Issue 7, pp. 2267-2286, DOI: 10.1007/s11269-013-0287-2.

Kisi, O. (2007). "Streamflow forecasting using different artificial neural network algorithms." *Journal of Hydrologic Engineering*, Vol. 12, Issue 5, pp. 532-539, DOI: 10.1061/(ASCE)1084-0699(2007)12:5(532).

Mallat, S. G. (1989). "A theory for multiresolution signal decomposition: The wavelet representation." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, Issue 7, pp. 674-693, DOI: 10.1109/34.192463.

Mamdani, E. H., and Assilian, S. (1975). "An experiment in linguistic synthesis with a fuzzy logic controller." *International Journal of Man-Machine Studies*, Vol. 7, Issue 1, pp. 1-13, DOI: 10.1016/S0020-7373(75)80002-2.

MathWorks (2014). *Fuzzy logic toolbox user's guide*, The MathWorks, Inc. http://www. mathworks.com/help/pdf_doc/fuzzy/fuzzy.pdf, (accessed 6 July 2014).

Mishra, A., Hata, T., and Abdelhadi, A. W. (2004). "Models for recession flows in the upper Blue Nile River." *Hydrological Processes*, Vol. 18, Issue 15, pp. 2773-2786, DOI: 10.1002/hyp.1322.

Mukerji, A., Chatterjee, C., and Raghuwanshi, N. S. (2009). "Flood forecasting using ANN, Neuro-Fuzzy, and Neuro-GA models." *Journal of Hydrologic Engineering*, Vol. 14, Issue 6, pp. 647-652, DOI: 10.1061/(ASCE)HE.1943-5584.0000040.

Nason, G. (2010). *Wavelet methods in statistics with R*, Springer, New York.

Nayak, P. C., Sudheer, K. P., Rangan, D. M., and Ramasastri, K. S. (2005). "Short-term flood forecasting with a neurofuzzy model." *Water Resources Research*, Vol. 41, Issue 4, W04004, *DOI: 10.1029/2004WR003562.*

Nejad, F. H., and Nourani, V. (2012). "Elevation of wavelet denoising performance via an ANN-based streamflow forecasting model." *International Journal of Computer Science and Management Research*, Vol. 1, Issue 4, pp. 764-770.

Nguyen, P. K. T., and Chua, L. H. C. (2012). "The data-driven approach as an operational real-time flood forecasting model." *Hydrological Processes*, Vol. 26, Issue 19, pp. 2878-2893, DOI: 10.1002/hyp.8347.

Nourani, V., Alami, M. T., and Aminfar, M. H. (2009). "A combined

neural-wavelet model for prediction of Ligvanchai watershed precipitation." *Engineering Applications of Artificial Intelligence*, Vol. 22, Issue 3, pp. 466-472, DOI: 10.1016/j.engappai.2008.09.003.

Nourani, V., Kisi, O., and Komasi, M. (2011). "Two hybrid artificial intelligence approaches for modeling rainfall-runoff process." *Journal of Hydrology*, Vol. 402, Issues 1-2, pp. 41-59, DOI: 10.1016/j.jhydrol.2011.03.002.

Okkan, U. (2012). "Using wavelet transform to improve generalization capability of feed forward neural networks in monthly runoff prediction." *Scientific Research and Essays*, Vol. 7, No. 17, pp. 1690-1703, DOI: 10.5897/SRE12.110.

Patel, D., and Parekh, D. F. (2014). "Flood forecasting using adaptive neuro-fuzzy inference system (ANFIS)." *International Journal of Engineering Trends and Technology*, Vol. 12, No. 10, pp. 510-514, DOI: 10.14445/22315381/IJETT-V12P295.

Piotrowski, A., Napiórkowski, J. J., and Rowiński, P. M. (2006). "Flash-flood forecasting by means of neural networks and nearest neighbor approach – a comparative study." *Nonlinear Processes in Geophysics*, Vol. 13, pp. 443-448, DOI: 10.5194/npg-13-443-2006.

Rezaeianzadeh, M., Tabari, H, Yazdi, A. A., Isik, S., and Kalin, L. (2014). "Flood flow forecasting using ANN, ANFIS and regression models." *Neural Computing and Applications*, Vol. 25, Issue 1, pp. 25-37, DOI: 10.1007/s00521-013-1443-6.

Salas, J. D., Delleur, J. W., Yevjevich, V., and Lane, W. L. (1985). *Applied modeling of hydrologic time series*, Water Resources Publications, Littleton, Colorado, USA.

Sehgal, V., Sahay, R. R., and Chatterjee, C. (2014). "Effect of utilization of discrete wavelet components on flood forecasting performance of wavelet based ANFIS models." *Water Resources Management*, Vol. 28, Issue 6, pp. 1733-1749, DOI: 10.1007/s11269-014-0584-4.

Seo, Y., Kim, S., and Singh, V. P. (2013a). "Flood forecasting and uncertainty assessment using bootstrapped ANFIS." *Proc. 6th Conf. of Asia Pacific Association of Hydrology and Water Resources*, Seoul, South Korea, pp. 1-8.

Seo, Y., Park, K. B., Kim, S., and Singh, V. P. (2013b). "Application of bootstrap-based artificial neural networks to flood forecasting and uncertainty assessment." *Proc. 6th International Perspective on Water Resources and the Environment*, EWRI-ASCE, Izmir, Turkey.

Sudheer, K. P., Gosain, A. K., and Ramasastri, K. S. (2002). "A data-driven algorithm for constructing artificial neural network rainfall-runoff models." *Hydrological Processes*, Vol. 16, Issue 6, pp. 1325-1330, DOI: 10.1002/hyp.554.

Takagi, T. and Sugeno, M. (1985). "Fuzzy identification of systems and its application to modeling and control." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-15, Issue 1, pp. 116-132, DOI: 10.1109/TSMC.1985.6313399.

Tiwari, M. K. and Chatterjee, C. (2010a). "Uncertainty assessment and ensemble flood forecasting using bootstrap based artificial neural networks (BANNs)." *Journal of Hydrology*, Vol. 382, Issues 1-4, pp. 20-33, DOI: 10.1016/j.jhydrol.2009.12.013.

Tiwari, M. K. and Chatterjee, C. (2010b). "Development of an accurate and reliable hourly flood forecasting model using wavelet-bootstrap-ANN (WBANN) hybrid approach." *Journal of Hydrology*, Vol. 394, Issues 3-4, pp. 458-470, DOI: 10.1016/j.jhydrol.2010.10.001.

Toth, E., Montanari, A., and Brath, A. (1999). "Real-time flood forecasting via combined use of conceptual and stochastic models." *Physics and Chemistry of the Earth, Part B: Hydrology, Oceans and Atmosphere*, Vol. 24, Issue 7, pp. 793-798, DOI: 10.1016/S1464-1909(99)00082-9.

Zhang, G., Patuwo, B. E., and Hu, M. Y. (1998). "Forecasting with artificial neural networks: The state of the art." *International Journal of Forecasting*, Vol. 14, Issue 1, pp. 35-62, DOI: 10.1016/S0169-2070(97)00044-7.