Investigation on the Potential to Integrate Different Artificial Intelligence Models with Metaheuristic Algorithms for Improving River Suspended Sediment Predictions

Mohammad Ehteram [1], Samira Ghotbi [2], Ozgur Kisi [3] , Ali Najah Ahmed [4],* ,
Gasim Hayder [4] , Chow Ming Fai [4,5], Mathivanan Krishnan [5], Haitham Abdulmohsin Afan [6]
and Ahmed EL-Shafie [6],*

[1] Department of Water Engineering and Hydraulic Structures, Faculty of Civil Engineering, Semnan University, Semnan 3513119111, Iran; mohammdehteram@semnan.ac.ir

[2] Department of Civil Engineering, Sharood University of Technology, Sharood 3616713455, Iran; sami_ghotbi@yahoo.com

[3] Faculty of Natural Sciences and Engineering, School of Technology, Ilia State University, Tbilisi 0162, Georgia; ozgur.kisi@iliauni.edu.ge

[4] Institute of Energy Infrastructure (IEI), Universiti Tenaga Nasional (UNITEN), Kajang, Selangor 43000, Malaysia; Gasim@uniten.edu.my (G.H.); Chowmf@uniten.edu.my (C.M.F.)

[5] Civil Engineering Department, College of Engineering, Universiti Tenaga Nasional (UNITEN), Kajang, Selangor 43000, Malaysia; mathivanan.krishnan@yahoo.com

[6] Civil Engineering Department, Faculty of Engineering, University of Malaya, Kuala Lumpur 50603, Malaysia; haitham.afan@gmail.com

* Correspondence: mahfoodh@uniten.edu.my (A.N.A.); elshafie@um.edu.my (A.E.-S.)

check for updates

Abstract: Suspended sediment load (SLL) prediction is a significant field in hydrology and hydraulic sciences, as sedimentation processes change the soil quality. Although the adaptive neuro fuzzy system (ANFIS) and multilayer feed-forward neural network (MFNN) have been widely used to simulate hydrological variables, improving the accuracy of the above models is an important issue for hydrologists. In this article, the ANFIS and MFNN models were improved by the bat algorithm (BA) and weed algorithm (WA). Thus, the current paper introduces improved ANFIS and MFNN models: ANFIS–BA, ANFIS–WA, MFNN–BA, and MFNN–WA. The models were validated by applying river discharge, rainfall, and monthly suspended sediment load (SSL) for the Atrek basin in Iran. In addition, seven input groups were used to predict monthly SSL. The best models were identified through root-mean-square error (RMSE), Nash–Sutcliff efficiency (NSE), standard deviation ratio (RSR), percent bias (PBIAS) indices, and uncertainty analysis. For the ANFIS–BA model, RMSE and RSR varied from 1.5 to 2.5 ton/d and from 5% to 25%, respectively. In addition, a variation range of NSE was between very good and good performance (0.75 to 0.85 and 0.85 to 1). The uncertainty analysis showed that the ANFIS–BA had more reliable performance compared to other models. Thus, the ANFIS–BA model has high potential for predicting SSL.

1. Introduction

Through basin management, different strategies have been applied to decrease sediment volume. Sedimentation processes and soil erosion are important in the fields of hydrology and hydraulics [1]. Sediment load relies on rainfall and runoff, as these hydrological variables can change the sedimentation

process. Sedimentation can decrease crop yield, available water volume, and soil quality [2]. Sediment load can also lead to pollution of the environment due to the transport of pollutants; thus, it is necessary to predict sediment load. Additionally, understanding the sedimentation process is important for the protection of hydraulic and flood structures [3]. Accurate prediction and estimation of sediment load is necessary for water resource management, flood control, water quality, and so on. [4]. In addition, other hydrological variables have a significant impact on sediment load prediction. Thus, application of accurate prediction models is necessary for the planning and management of basins. Soft computing models have been widely applied to simulate hydrological variables [5]. These models are applied for forecasting the climate variables in different fields. By using soft computing models, model accuracy can be enhanced [6]. Soft computing models can present conceptions of the hydrological process but need a large set of inputs. With soft computing models, outputs are obtained without in-depth observation of the physical details of the hydrological issues being investigated. A key benefit of soft computing models is that they can learn the system behavior when hydrological inputs are inserted into the model. The other benefits of applying soft computing models are fast computation, high accuracy, high flexibility, and a simple computation process. The investigated literature reviews emphasize that the preparation of soft computing models has unknown parameters [7–9]. These studies show that the training of adaptive neuro fuzzy system (ANFIS) parameters is an important challenge during the simulation process. Although there are some training algorithms, such as gradient methods, for obtaining ANFIS parameters, the computation of gradients in each level is very sophisticated. In addition, training of artificial neural network (ANN) model parameters is a main challenge. For example, the number of neurons in hidden layer(s), weight values, and biases are important parameters for the ANN models [10].

An effective optimization algorithm is necessary for improving soft computing models. There are different soft computing models, but high skill is required to determine the unknown parameters of such models in order to improve their performance. An accurate investigation is also necessary to select the best architecture of ANFIS and multilayer feed-forward neural network (MFNN) models [11]. It should be mentioned that although the ANFIS and MFNN models have high performances among other simulation models, their accuracies depend on accurate preparation of their parameters. Thus, new optimization algorithms can be applied for updating and training of the ANFIS and ANN parameters [12]. Thus, different studies attempted to use new training strategies that are more accurate and easier than the traditional methods for obtaining model parameters.

Developing efficient models for suspended load prediction is the important objective of this article. The different steps of this article are as follows:

1. To improve ANFIS and MFNN model efficiency by applying new optimization algorithms. These algorithms are used to obtain the best ANFIS and MFNN structures and parameters;
2. To predict monthly sediment load by applying improved ANFIS and MFNN models;
3. To examine the uncertainty of the predictions; and
4. To obtain a sediment map of the case study.

2. Literature Review

Guven and Kisi [13] applied a machine learning genetic programming (MLGP) model to obtain daily-suspended load. It was observed that the MLGP model outperformed the genetic expression programming (GEP) model. The prediction error, root-mean-square error (RMSE), obtained from the MLGP model was 175 ton/d, while for the GEP model, the RMSE was 231 ton/d.

Kisi et al. [14] compared the genetic programming (GP) model with the support vector machine (SVM) and ANN models in daily-suspended load prediction. The results indicated that the GP model outperformed the SVM and ANN models.

Chaing et al. [15] compared the SVM, ANN, and linear regression models to predict suspended sediment load. The results revealed that the SVM model outperformed the ANN and linear regression

models. They applied 85% of the input data for training the soft computing models, and the remaining 15% of the data were applied for testing.

Another paper studied the capabilities of SVM and ANN models to predict daily-suspended load. The suspended load was modeled by using the inputs of rainfall and streamflow. A gamma test was used to select input data [16]. It was found that the SVR model with radial basis kernel function presented more accurate outputs than other models.

Another study investigated the performance of the ANN model in forecasting sediment movement by applying the assumptions of self-filtering sewer models [17]. It was found that the ANN model outperformed SVM and GP models. Velocity and shear stress were used as input data.

Rainfall, discharge, and antecedent sediment load were used as input to the ANN model [18]. Outputs from the ANN model were benchmarked against multiple linear regression (MLR). The reported ANN model could be effectively applied to decrease the frequency of high-priced operations for sediment evaluations.

Shiau and Chen [19] applied a probabilistic system for forecasting daily-suspended loads. The accuracy of the probabilistic model was better than experimental rating curve methods. Statistical characteristics such as median, mean, and mode of the obtained probability distribution were the model outputs.

Chen and Chau [20] applied a hybrid double-forward neural network (HDFNN) model to predict suspended load. The obtained peak value of suspended sediment load (SSL) showed that the HDFNN model was more suitable than the multilayer feed-forward neural network.

A staking method was used to forecast SSL [21]. To investigate the suggested method, two case studies from the USA were considered. Streamflow and suspended sediment concentration were model inputs. The estimated outputs illustrated that the suggested method performed better than genetic programming and regression models.

In a study in Turkey, the capabilities of SVM, ANN, and ANFIS models were studied to predict suspended load [22]. The simulation was made by applying two input variables, streamflow, and suspended sediment load. Results showed that the ANFIS model could give a lower RMSE than the other models.

To estimate SSL, a multilayer perceptron (MLP) neural network model was trained by different algorithms [23]. The results indicated that the Levenberg–Marquardt algorithm could reach fast convergence. The performance of a scaled conjugate gradient (SCG) algorithm had the next priority.

Hybrid tree models were used to forecast SSL [24]. Water temperature, water discharge, and electrical conductivity were model inputs. The Friedman test was applied to determine the best models. The results indicated that the hybrid tree models outperformed simple tree models. The random subspace error pruning (RSEP) model performed worse than other models.

Kisi and Yaseen [14] applied the evolutionary fuzzy (EF) method to forecast SSL. The prediction model was created by applying discharge data. The obtained results proved the high capability of the EF model for SSL prediction. They suggested that the model could be a useful tool for river engineering.

Another study investigated the ability of the ANN method for modeling sophisticated nonlinear SSL in the Himalayan area [25]. The research not only presented an understanding of the sedimentation process, but it also studied the impact of other hydrological variables on SSL.

A three-layer back propagation model was used to estimate SSL. Rainfall, water discharge, and sediment discharge were model inputs [4]. To investigate the ability of the model, error was computed by benchmarking the observed and simulated data. It was found that the model could present accurate outputs in the prediction of SSL.

In general, the literature review showed that soft computing models are widely used for SSL prediction. However, soft computing models can be improved by combining them with optimization algorithms. In the current study, two new optimization algorithms are used to improve the ANFIS and MFNN models. Bat algorithm (BA) and weed algorithm (WA) are widely used in a number of different issues, including water resource management, water quality, hydraulic structure design, mathematical problems, power generation, hydrological simulations, and flood control [26,27]. These

algorithms are widely used because of their fast convergence, simple computation process, good balance between exploration and exploitation abilities, high flexibility, accurate results, and the high capability of local and global searches. To our knowledge, the hybrid ANFIS–BA, MFNN–BA, ANFIS–WA, and MFNN–WA models have not been previously used for SSL prediction.

The current study not only predicts monthly SSL using improved ANFIS and MFNN models, but it also presents comprehensive information about the uncertainty of the prediction models. In addition, a comprehensive evaluation is carried out to study the advantages and disadvantages of the applied soft computing models.

2.1. Case Study

The Atrek River in Iran is located in the southeastern district of the Caspian Sea. This basin is located geographically at 56′01′32″ to 56′01′32″ E and 36°56′32″ to 38°14′42″ N, as shown in Figure 1. The Atrek basin (14,931.24 km²) is one of the largest basins in Iran. The climate of this basin is semiarid, with a cold winter and hot summer. The average annual temperature and precipitation are 13.3 °C and 235.3 mm, respectively. The soil type of the basin is classified into seven groups that contain 38.8% loam–clay sand, 22.4% sand, 21.3% loam–sand, and 12.8% sand–loam, and the remaining 9.7% data represent loam, loam–clay, and loam–silt. The watershed consists of mountains and flat areas. Its height is 378 to 2823 m above sea level. Sediment transport from adjacent beaches, coastal areas, and shore face deposits are important sources of sedimentation processes in this basin. In addition, human activities change soil, vegetation coverage, and land use. Thus, in this basin, human activities are one of the most important reasons for the sedimentation process. Rainfall, discharge, and SLL data from 1997 to 2017 were used. The information was classified in two sets: a training data set (1997–2011) and a testing data set (2012–2017). The trained models used the discharge and rainfall.

Figure 1. Atrek Basin.

Figure 2 shows average monthly rainfall. The months of April (range of rainfall: 30–150 mm) and October (range of rainfall: 12–16 mm) have the highest and lowest rainfall uncertainty among all the seasons. Spring is the wettest season, averaging 98 mm. The minimum rainfall is 24 mm in the summer. In addition, the months of June and March have the highest and lowest temperature uncertainty (Figure 3). As observed from Figures 2 and 3, the temperature and rainfall have high variations. This shows a complex relationship between sediment load and hydrological variables. The measured discharges are shown in Figure 4. The Spring season has the most discharge variability. The months of March and July have the highest and lowest discharge. Importantly, significant variability in the temporal and spatial distribution of temperature, rainfall, and discharge presented here has significant

effects on SSL prediction. Thus, the study of temporal and spatial variability of SSL is important to enhance knowledge of the sedimentation process. The discharge variations are considered as inputs, and their effects are evaluated on the outputs or sediment load.
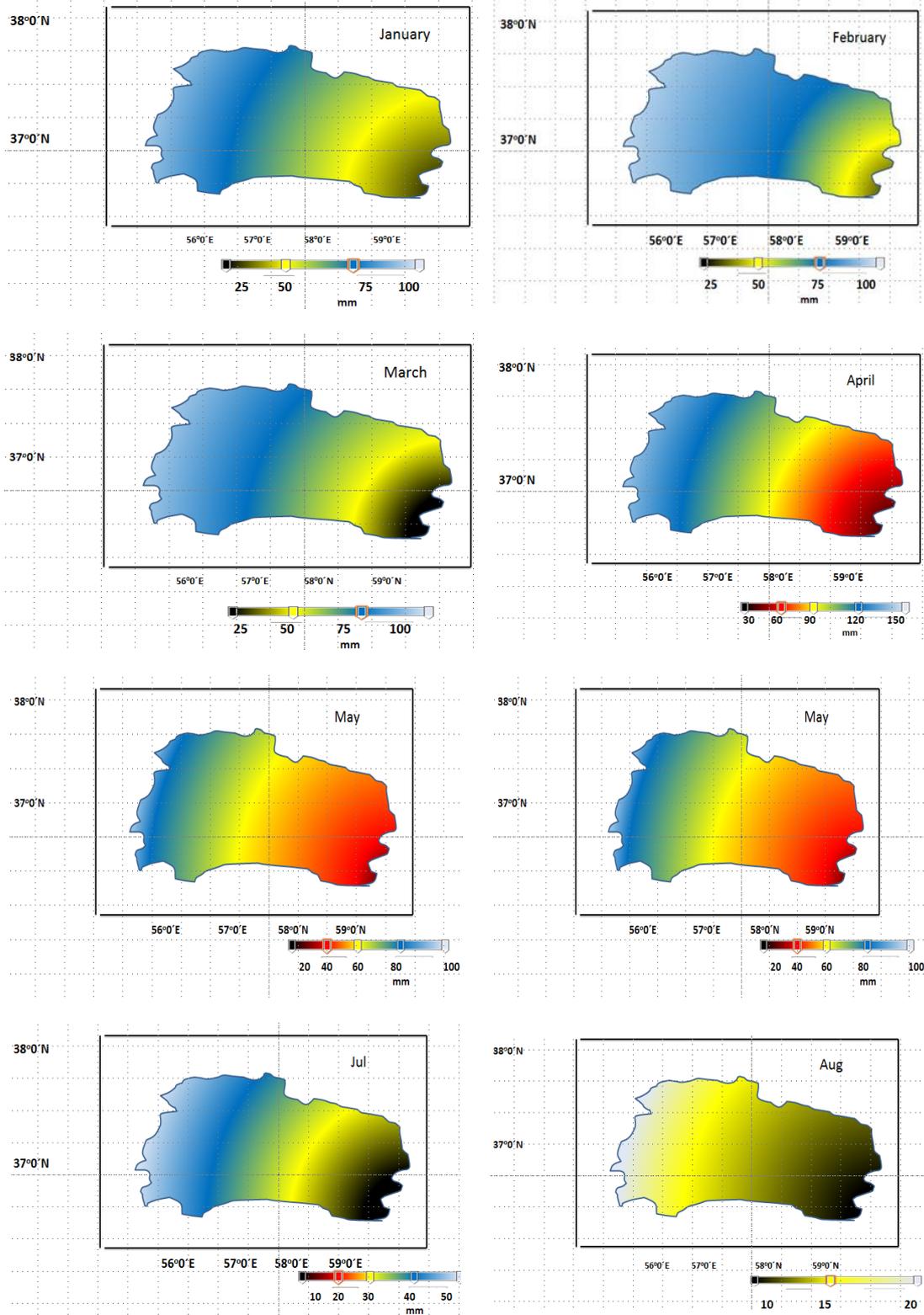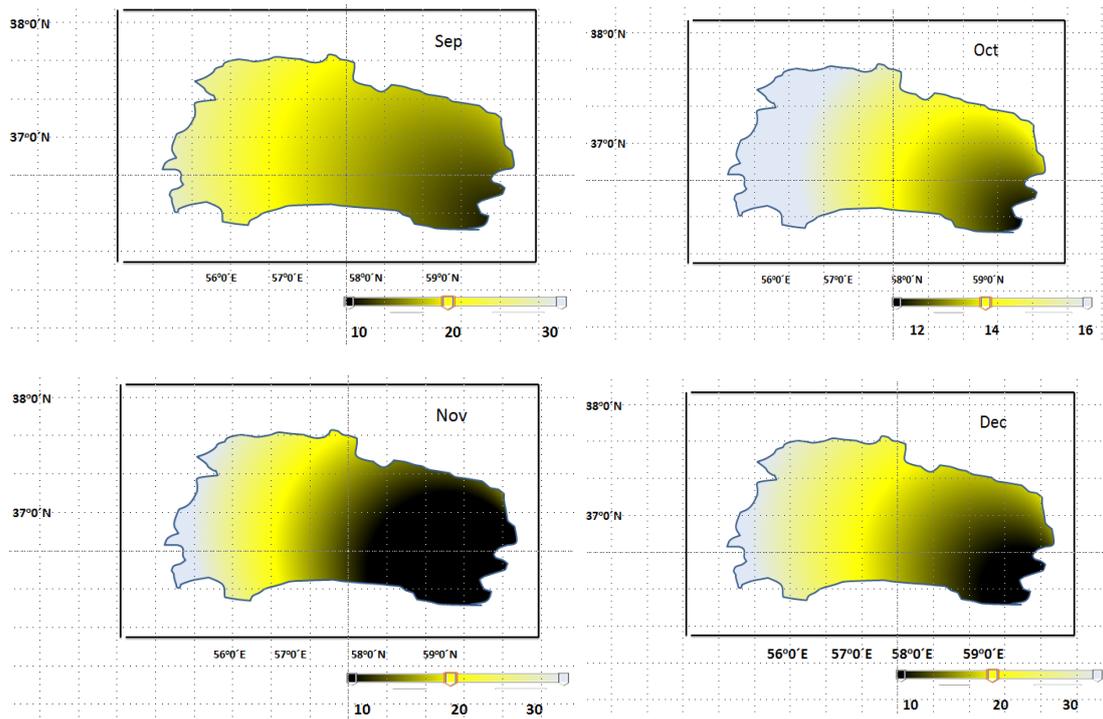

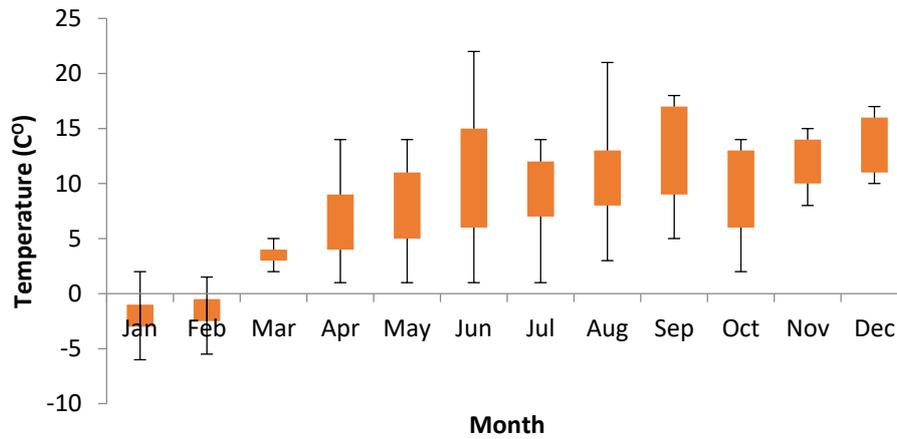**Figure 2.** *Cont.*

Figure 2. Monthly rainfall for the basin.

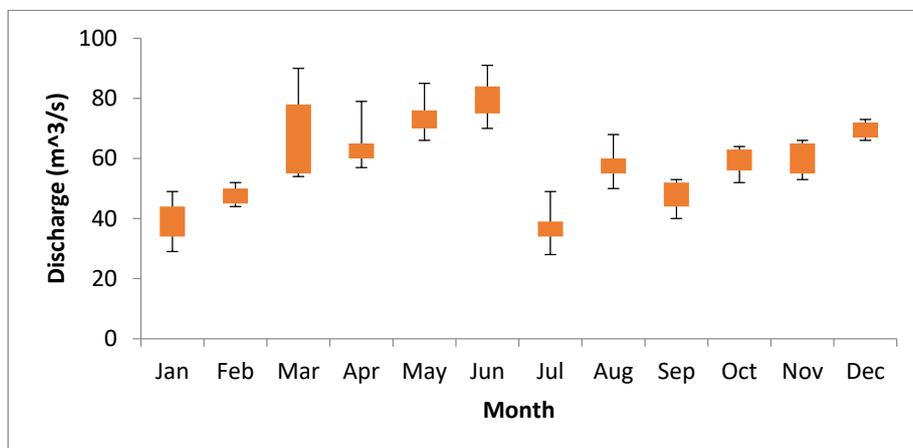**Figure 3.** Monthly temperature for the basin.

Figure 4. Monthly discharge for the basin.

2.2. Data and Parameters Applied in Suspended Sediment Load (SSL) Prediction

The information applied here to forecast SSL in the Atrek basin contains monthly SSL, monthly discharge, and precipitation during 1997–2017. The data were obtained from the Iran water resource management organization. Seven groups of input data were applied for improved ANFIS and MFNN models. A lag time of 3 to 9 months was considered for SSL prediction. The best inputs were obtained by principal component analysis (PCA). In the case of hydrological predictions, accurate knowledge of effective predictors is important for decision makers [28]. PCA is an identification method that applies orthogonal transformation to turn a collection of measurements of conceivably associated parameters into a collection of values of linearly unassociated parameters named principal components. However, there are min (m^{-1}, z) components when m observations and z variables are defined for data. When a component has the largest variance, the considered component is the most effective component. PCA is widely applied to determine the effective inputs for estimation of hydrological variables. Therefore, the proposed structures for the model inputs–outputs pattern were obtained from PCA, as shown in Table 1. The proposed models have been developed using Matlab 9.5 software version R2018b (MathWorks, USA) with a system 17 processor, SSD, and 64 GB RAM. On the other hand, the PCA computations have been carried out using SPSS 2018 software version 25.0.0.0 (IBM Corporation, USA). All the information and the data used in this study have been collected from water resource management authorities in Iran.

Table 1. The suggested combination.

Model Simulation	Parameter	Time Lag	Output
Group 1	SSL	t-3	Suspended Sediment Load (SSL) (t)
Group 2	Q, SSL	t-3	SSL (t)
Group 3	SSL, R	t-3	SSL (t)
Group 4	Q, R	t-3	SSL (t)
Group 5	SSL, Q, R	t-3	SSL (t)
Group 6	SSL, Q, R	t-6	SSL (t)
Group 7	SSL, Q, R	t-9	SSL (t)

2.3. Adaptive Neuro Fuzzy System (ANFIS) Method

The structure of the ANFIS model is shown in Figure 5. There are five layers to this system [29]:

1. The first layer calculates the membership degree. Each node produces a membership degree. The fuzzy sets apply membership functions [29].

$$O_{1,i} = \mu_{Ai}(x)i = 1, 2, \ldots$$
$$O_{1,i} = \mu_{Bi-2}(y)i = 3, 4, \ldots \tag{1}$$

   where x and y are outputs, A_i and B_i are linguistic labels, and μ_{Ai} and μ_{Bi-2} are the degree of membership function for A_i and B_i, respectively.

2. The output of the second layer (fire strengths) is computed based on the computed membership degrees. In fact, membership functions of the previous layer are compounded together to generate the firing strengths.

$$O_{2i} = w_i = \mu A_i(x) * \mu B_i(y), \tag{2}$$

   where O_{2i} is the output of this layer called fire strength.

3. The firing strengths are normalized in this level. The contribution of the firing strengths is computed by the constant nodes:

$$O_{3,i} = w_i = \frac{W_i}{w_1 + w_2}. \tag{3}$$

4.　In this layer, consequent parameters are used to determine the proportion of the *i*th rule to the overall outcomes:

$$O_{4,i} = \overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i), \tag{4}$$

where p_i, q_i, and r_i are consequent parameters.

5.　This layer uses summation of input signals to obtain the overall output.

$$O_{5,i} = \sum_i \overline{w} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}. \tag{5a}$$

A previous study showed that a Gaussian membership function led to accurate outputs [30]. Thus, the current study uses this membership function [29]:

$$\mu_{A_i}(x) = \left(-\frac{(x - c_i)^2}{2\sigma_i 2} \right), \tag{5b}$$

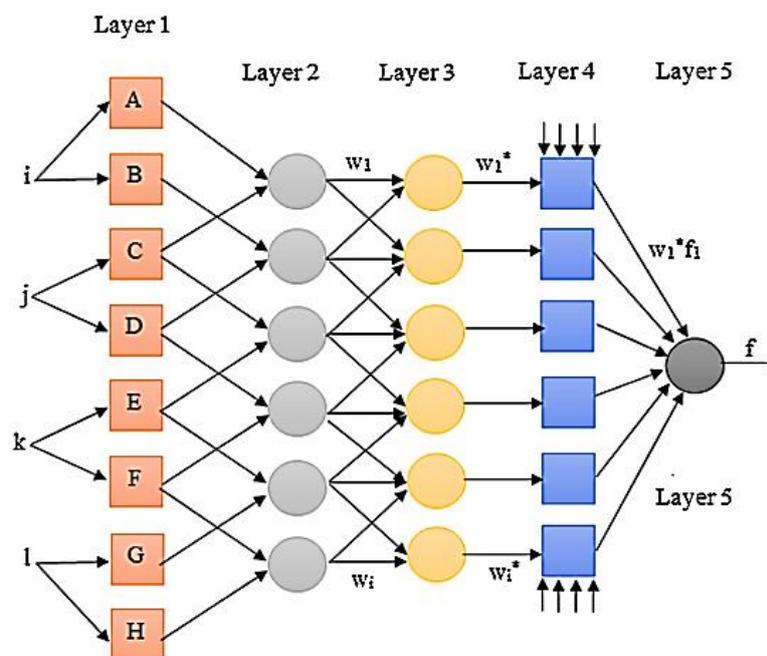where c_i and σ_i are the parameters for the membership function.

Figure 5. Adaptive neuro fuzzy system (ANFIS) model.

2.4. Feed-Forward Neural Network (FNN)

The MFNN was used for the current study. This method is widely used for predicting different hydrological variables. This method generally includes three layers between inputs and outputs [31]. The first layer includes Z inputs ($z = 1, 2, 3, \ldots, Z$). The second layer is known as the hidden layer and includes Q inputs ($q = 1, 2, 3, \ldots, Q$). The weight parameters (w_{zk}, $z = 1, 2, 3, \ldots, Z$, $k = 1, 2, \ldots, K$) and (w_{qk}, $q = 1, 2, 3, \ldots, Q$, $k = 1, 2, \ldots, K$) connect the input layer to the hidden layer and the hidden

layer to the output layer, respectively. Each input neuron to the hidden layer (t_q) generates an output (v_q). d_k and y_q are the input neuron and total output of the output layer.

$$
\begin{aligned}
t_q &= \sum_{z=1}^{Z} w_{zq} x_z, v_q = f[t_q] = f\left[\sum_{z=1}^{Z} w_{zq} x_z\right] \\
d_k &= \sum_{q=1}^{Q} w_{qk} v_q, y_q = f[d_k] = f\left[\sum_{q=1}^{Q} w_{qk} v_q\right]
\end{aligned}
\tag{6}
$$

where $f(.)$ determines the transfer function. This function is used to investigate the connections between the inputs and outputs.

2.5. ANFIS and Multilayer FNN (MFNN) Models and Optimization Algorithms

While the ANFIS model has a good ability to obtain the output from the inputs as a soft computing model, a long and complex computation process is required to find the best values of the consequent and MF (membership function) parameters. The MFNN model has a few unknown components: number of neurons in the hidden layer weights, biases, and activation functions (either per layer or per neuron).

Thus, the optimization algorithms are used for computing ANFIS and MFNN parameters. A comprehensive explanation is presented for each algorithm based on the following parts:

1. Bat algorithm (BA): BA is an optimization algorithm that acts based on echolocation characteristics. Each of the flying bats has a random velocity and flies at random positions. When it is searching for prey, its loudness, frequency, and pulsation rates are widely varied. A local random walk is used to increase the search ability [26]. The velocity and position components are used to obtain the optimal solutions. Each position is a candidate solution. Thus, the best position is the best solution while the algorithm should avoid trapping in the local optimums (Figure 6). In fact, bats generate sounds that are returned from the surroundings. They can differentiate an obstacle from prey based on returned frequencies [26]. The velocity, frequency, and position are defined as:

$$
f_l = f_{\min} + (f_{\max} - f_{\min}) \times \beta,
\tag{7}
$$

$$
v_l(t) = [x_l(t) - x_*] \times f_l,
\tag{8}
$$

$$
x_l(t) = x_l(t-1) + v_l(t),
\tag{9}
$$

where f_l is frequency, $f_{\min}$ is minimum frequency, $f_{\max}$ is maximum frequency, $v_l(t)$ is velocity, $x_l(t)$ is position, and x_* is the best position. Equation (9) defines the local search applying a random walk [30]:

$$
x(t) = x(t-1) + \varepsilon A_t,
\tag{10}
$$

where A_t is loudness and ε is a random value. The pulsation and loudness rate are varied during computational levels. If a bat finds its prey, the pulsation (r_l) and loudness rate have an increasing and decreasing trend, respectively. The quality solutions are identified by the objective function computation (Figure 6).

2. Weed algorithm (WA): Weeds attempt to find the best growth position. Among all the characteristics of this optimization algorithm, its easy structure, high accuracy, and few random parameters make it useful for practice in optimization problems.

    WA has different levels as follows [27]:

*   Initialization: The initial population is distributed randomly in a d dimensional problem space. The locations of weeds are considered as the decision-making components.
*   Reproduction: The weeds generate a particular number of seeds. The number of seeds varies from $S_{\min}$ (minimum number of seeds) to $S_{\max}$ (maximum number of seeds). The number of seeds is

S_{max} if a weed has the best objective function value. Although the quality of some weeds is not good, the reproduction process allows them to have a chance again for continuation of life. This issue is important because some of them may have important information. If they do not have another chance, important information is eliminated from the algorithm cycle.

- Competitive level: The combination of weeds produces the next weed generation. If the population exceeds a threshold, the weeds with low quality are eliminated compared to weeds with high quality.

- Termination level: The algorithm finishes when the number of algorithm levels of iteration (Iter) reaches the maximum number of iterations (iter-max), which has been proposed to be 1000 iterations. (Figure 7).
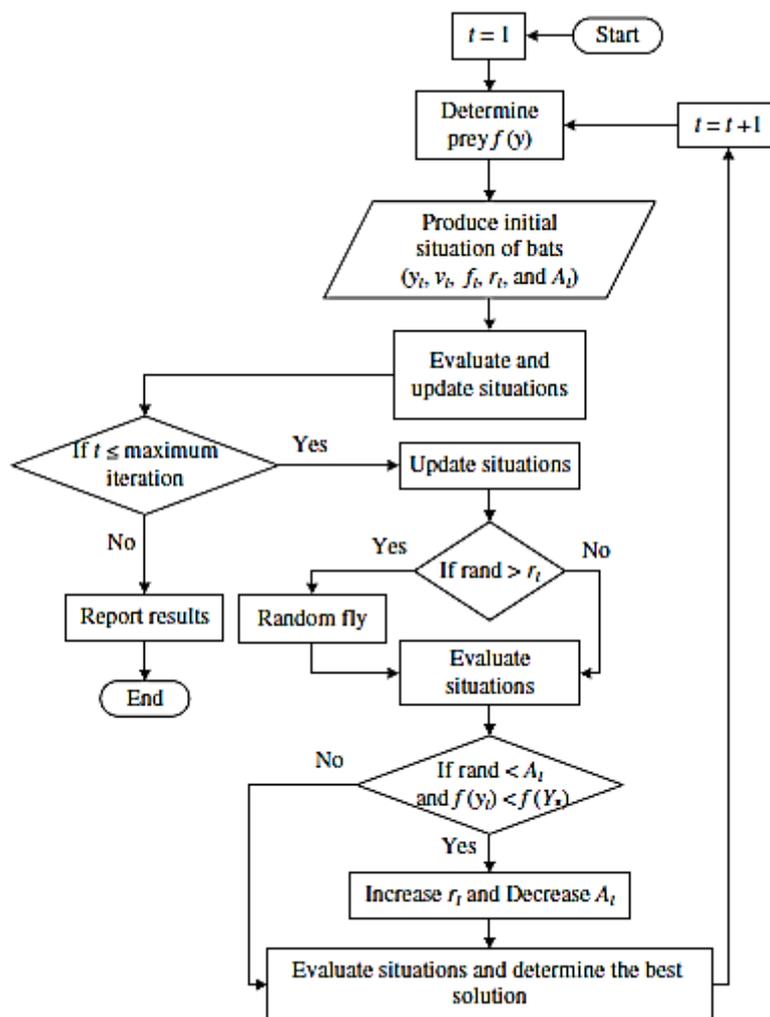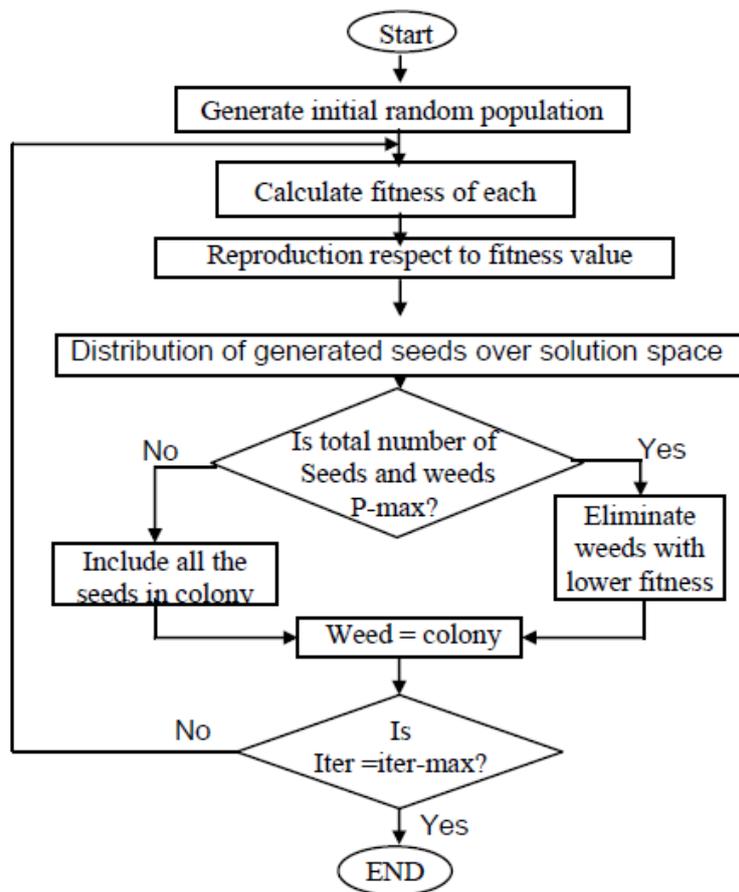

Figure 6. Structure of the bat algorithm (BA).

Figure 7. Structure of the weed algorithm (WA).

Figure 8 shows the simulation process based on the ANFIS and optimization algorithms.
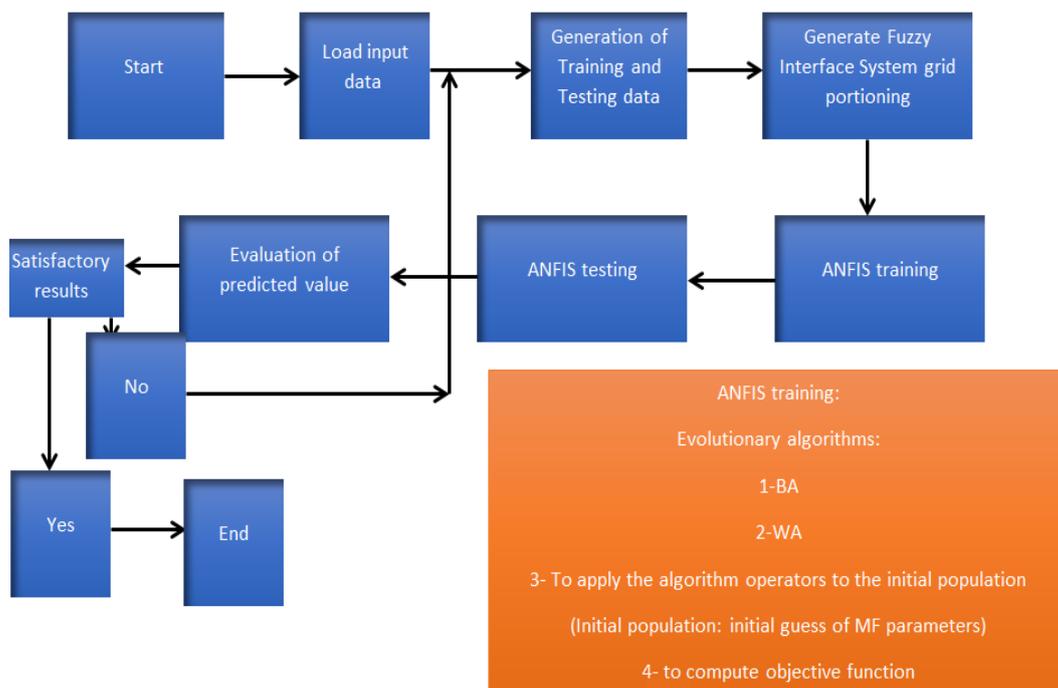

Figure 8. ANFIS and optimization algorithm.

The following indices are used to evaluate different models [31,32]:

1.   Percent bias:

$$PBIAS = \left[\frac{\sum\limits_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim}\right) * 100}{\sum\limits_{i=1}^{n} \left(Y_i^{obse}\right)} \right].$$ (11)

2.   Standard deviation of RMSE observations

$$RSR = \frac{RMSE}{STDEV_{obs}} = \frac{\left[\sqrt{\sum\limits_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim}\right)^2} \right]}{\left[\sqrt{\sum\limits_{i=1}^{n} \left(Y_i^{obs} - Y_i^{mean}\right)^2} \right]}.$$ (12)

3.   Nash–Sutcliff efficiency

$$NSE = 1 - \left[\frac{\sum\limits_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim}\right)^2}{\sum\limits_{i=1}^{n} \left(Y_i^{obs} - Y^{mean}\right)^2} \right].$$ (13)

4.   RMSE (root-mean-square error)

$$RMSE = \sqrt{\frac{\sum\limits_{i=1}^{n} \left(Y_i^{obs} - Y_i^{sim}\right)^2}{N}}$$ (14)

Y_i^{obs} is observed data, Y_i^{sim} is simulated data, N is the number of data, and Y^{mean} is the average value of the data.

3. Results and Discussion

3.1. Sensitivity Analysis for Optimization Algorithms

To apply the optimization algorithms to the ANFIS and MFNN models, accurate adjustment of random parameters must be obtained a priori. As in the evolutionary algorithms, the BA and WA are fully sensitive to initialize random parameters. In fact, the random parameters change the accuracy of optimization algorithms. Thus, it is necessary to find the optimal value of random parameters so that the objective function values (RMSE in this study) converges to the lowest value.

Hence, to obtain better efficiency of the BA and WA, it is necessary to set the random parameters. As the WA and BA parameters were various, to create confidence that a suitable adjustment of random parameters was used, different experiments were used for generating the sensitivity analysis. The value of desired parameters was gradually changed, and the analysis was accomplished. The default values of other random parameters were constant when the user varied one parameter. The algorithm variations were examined based on the mean of the objective function value (i.e., RMSE). One sample of implanted sensitivity analysis to obtain proper adjustment of parameters systematically as the consecutive levels of needed analysis is shown in Table 2. In fact, Table 2 is a sample to show how to obtain the optimal value of random parameters. First, a population size, MaxF, MinF, MaxA, and MinA (1) were selected, and an objective function was computed for this combination. Then, MaxF, MinF, and MaxA with next value of MinA (3) were applied to compute the objective function value for the next combination. When all of the available MaxF, MinF, and MaxA with different values of MinF were generated, and the objective function value for each combination was generated, Max f was

changed to the next value (0.5). Then, the available combination was generated for the previous population size (10), MaxF (0.5), the previous MinF (0.1), and the previous MaxA (3) with MinA (1). Then, all available combinations for SP:10, MaxF (0.5), MinF (0.1), MaxA (3), and Min A from 1 to 4 were generated, and the objective function value was computed. This process was similarly repeated for the other population sizes and parameter values. Each combination had the lowest value for the objective function, and the best value of random parameters can be seen in Table 3. In fact, the parameters in Table 3 are the optimal values of random parameters that generated the lowest value of the objective function.

It should be mentioned that the best value of the parameters occurred at the lowest objective function value. RMSE or objective function had small error index values and were suitable for the simulations. Finally, Table 3 reports the best values of BA and WA parameters based on the above process.

Table 2. A sample of sensitivity analyses.

(a) STEP 1

SP: Size of Population	Maximum Frequency (MaxF)	Minimum Frequency (MinF)	Maximum Loudness (MaxA)	Minimum Loudness (MinA)
10	0.3	0.10	3	1
30	0.5	0.20	5	2
50	0.7	0.30	7	3
70	0.90	0.40	9	4

Mean objective function for the first combination of random parameters: 3.12 ton/month (SP:10, MAXF:0.30, MINF:0.10, MAXA:3 and Min A:1)

(b) STEP 2

SP: Size of Population	Maximum Frequency (MAXF)	Minimum Frequency (MinF)	Maximum Loudness (MAXA)	Minimum Loudness (MinA)
10	0.3	0.10	3	1
30	0.5	0.2	5	2
50	0.7	0.30	7	3
70	0.90	0.40	9	4

Mean objective function for the first combination of random parameters: 3.02 ton/month (SP:10, MAXF:0.30, MINF:0.10, MAXA:3 and Min A:3)

(c) STEP 3

SP: Size of Population	Maximum Frequency (MAXF)	Minimum Frequency (MinF)	Maximum Loudness (MAXA)	Minimum Loudness (MinA)
10	0.3	0.10	3	1
30	0.5	0.20	5	2
50	0.7	0.30	7	3
70	0.90	0.40	9	4

Mean objective function for the first combination of random parameters: 2.97 ton/month (SP:10, MAXF:0.30, MINF:0.10, MAXA:5 and Min A:3)

(d) STEP 4

SP: Size of Population	Maximum Frequency (MAXF)	Minimum Frequency (MinF)	Maximum Loudness (MAXA)	Minimum Loudness (MinA)
10	0.3	0.10	3	1
30	0.5	0.20	5	2
50	0.7	0.30	7	3
70	0.90	0.40	9	4

Mean objective function for the first combination of random parameters: 3.15 ton/month (SP:10, MAXF:0.30, MINF:0.10, MAXA:5 and Min A:7)

Table 3. The optimum values of BA and WA parameters.

BA	Parameter Value	WA	Parameter Value
SP	30	MAXSP	10
MAXF	0.7	Minimum SP	40
MINF	0.10	Maximum number of seeds	7
MAXA	5	Minimum number of seeds	1
MinA	1	Maximum iteration	900

3.2. Performance Index Analysis

The ANFIS–BA, ANFIS–WA, MFNN–BA, and MFNN–WA models were employed to predict monthly SSL. Figure 9 was generated based on the following performance indices: RMSE, Nash–Sutcliff efficiency (NSE), standard deviation ratio (RSR), and percent bias (PBIAS). In order to present the performances of these models, monthly SSL data were predicted utilizing the proposed models, and the monthly error values were computed based on the suggested performance indicator. Recall that the SSL predictions were achieved based not only on different model types but also on different input patterns as a model structure. Figure 9 was generated in this paper in order to simplify the comparison process and to present comprehensive information about the performance of each model. Since seven different model input combinations were considered, there were seven different error patterns for each model. The gradient charts were produced based on the lowest and highest error values of the attained errors, and then a classification for the error domain was considered. Finally, variations of error values were shown by varying the intensity of colors.
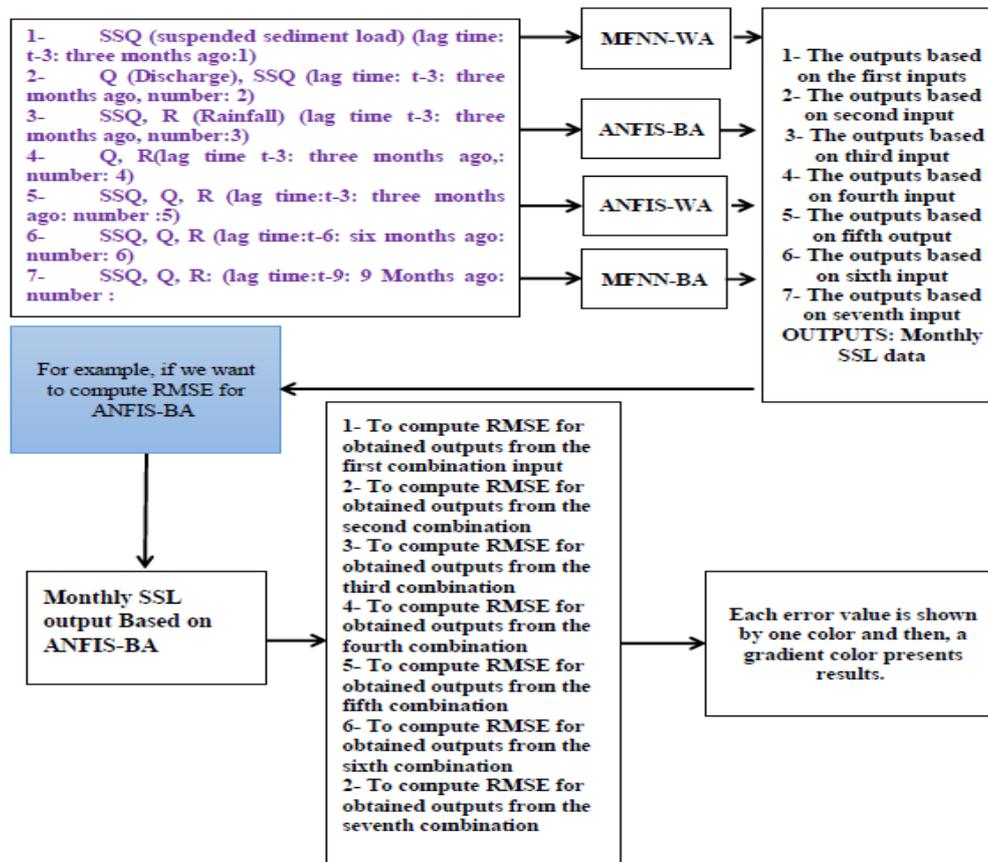

Figure 9. Computation process of errors.

In fact, Figure 9 was generated based on the application of each input combination and the computation of each index for each predicated SSL output. The vertical axis shows the number of input combinations:

1. SSL (suspended sediment load) (lag time t–3: three months ago, number: 1);
2. Q (discharge), SSQ (lag time t-3: three months ago, number: 2);
3. SSQ, R (rainfall) (lag time t-3: three months ago, number: 3);
4. Q, R (lag time t-3: three months ago, number: 4);
5. SSL, Q, R (lag time t-3: three months ago, number: 5);
6. SSL, Q, R (lag time t-6: 6 months ago, number: 6); and
7. SSL, Q, R: (lag time t-9: 9 months ago, number: 7).

The results for the models were compared with the observed data, and the statistical results were reported.

1. RMSE (Figure 10)

The range of RMSE values varied from 1.5 to 5.5 ton/d. The ANFIS–BA had the best results since RMSE varied from 1.5 to 3.5 to a large extent (Figure 10). For the seventh combination, SSQ, Q, and R (time lag: t-9), it can be understood that during testing and training levels, the ANFIS–BA resulted in minimal RMSE. For the first combination, SSQ (time lag: t-3), it can be understood that during testing and training levels, the ANFIS–BA had the weakest performance among the ANFIS–BA models with different combinations. For the ANFIS–WA, the range of RMSE values varied from 2.5 to 4.5 ton/d. After ANFIS–BA, with different combinations, ANFIS–WA had the next priority for predicting SSL. For the MFNN–WA, it had the weakest performance since RMSE varied from 3.5 to 5.5. It should be mentioned that during the testing level, the MFNN–WA resulted in maximal RMSE (5.5). In all models, the seventh and first combination had the best and weakest performances, respectively.
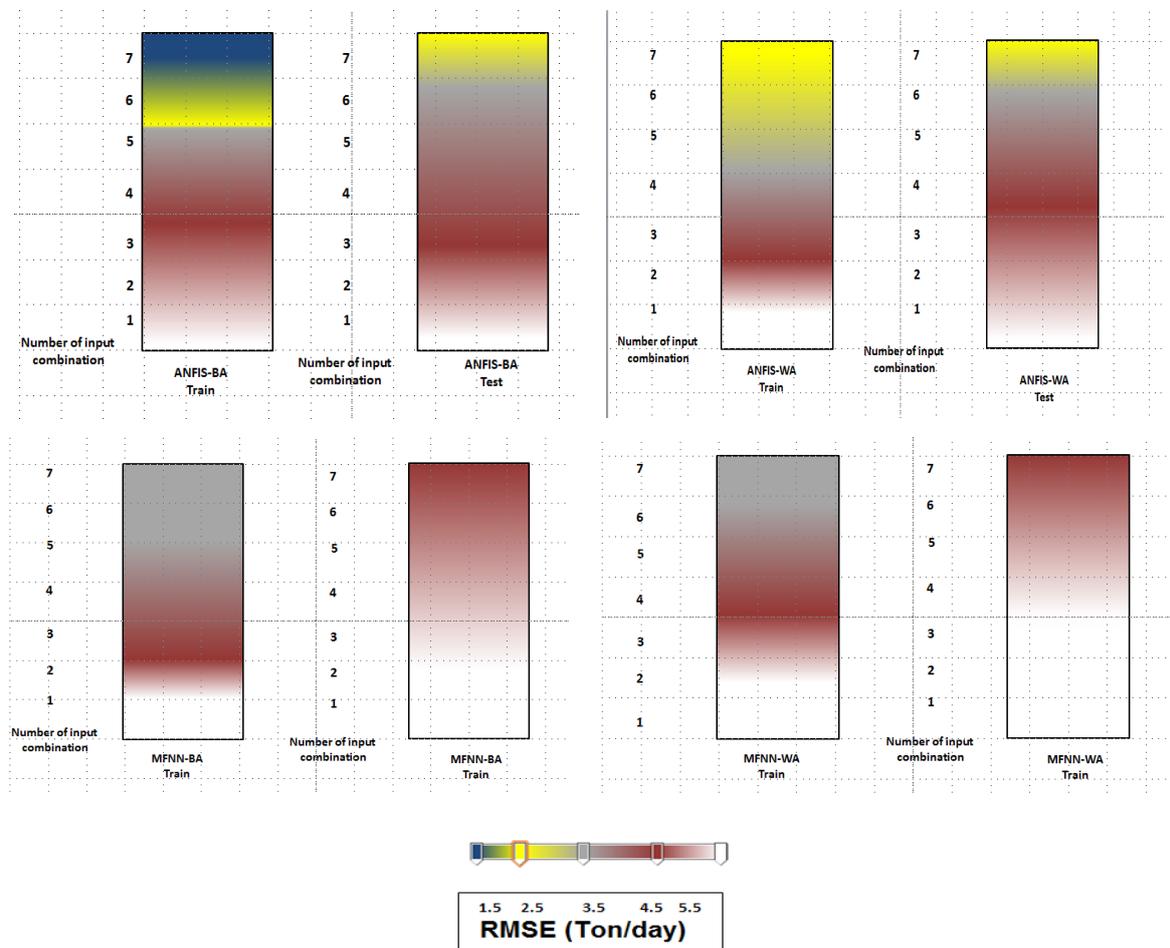

Figure 10. RMSE computations for different combinations.

2.   RSR (Figure 11)

RSR range varied from 0.05 to 0.25 (Figure 11). The small values showed the best performance of the models. For the ANFIS–BA, the RSR range varied from 0 to 0.60. Thus, it had the best performance among the models. After the ANFIS–BA, the ANFIS–WA and MFNN–BA had the best and weakest performances, respectively. The fifth, sixth, and seventh combinations had the same components, but the lag time was different for those models. It can be seen (Figure 9) that increasing the lag time improved the results for all models. As can be observed, the MFNN–BA model had better performance than the MFNN–WA model, since most of the results has smaller RSRs than that of the MFNN–WA model.

Figure 11. Standard deviation ratio (RSR) computations for different models.

3.   NSE and PBIAS (Figures 12 and 13)

There are general efficiency ratings for NSE and PBIAS as follows [20]:

$$\text{Very good}: \ 0.75 \leq NSE \leq 1.00, PBIAS < \pm 15,$$

$$\text{Good}: \ 0.65 \leq NSE \leq 0.75, \pm 15 \leq PBIAS < \pm 30,$$

$$\text{Satisfactory}: \ 0.50 \leq NSE \leq 0.65, \pm 30 \leq PBIAS < \pm 55, \text{ and}$$

$$\text{Unsatisfactory}: \ NSE \leq 0.50, PBIAS \geq \pm 55.$$

It can be seen that the performance of ANFIS–BA was very good and good. In fact, NSE and PBIAS were separately computed for each month and then classified based on these efficiency ratings. To a large extent, the MFNN–WA model had unsatisfactory results. However, the MFNN–BA had a

better performance than the other MFNN–WA models because the good and satisfactory ratings for the MFNN–BA model were more than those of the MFNN–WA model. Generally, the training level had a better performance than the testing level (Figures 12 and 13).

Previous studies showed the application of the ANFIS model with other optimization algorithms [15]. For example, a genetic algorithm was used to improve the ANFIS model, and SSL was simulated. In fact, the ANFIS parameters were considered as the initial population of chromosomes [33]. Another study used the ANFIS model with particle swarm optimization (ANFIS–PSO) [34]. In fact, the ANFIS parameters were considered as the particle positions. Then, the velocity and position were updated for each level. The results of the current paper were compared to the ANFIS–PSO and ANFIS–GA models developed in previous studies, as seen in Figure 14. The Taylor diagram shows better performance of the ANFIS–BA model compared to the ANFIS–PSO and ANFIS–GA models. The Taylor diagram is based on standard deviation, correlation coefficient, and normalized RMSE. The ANFIS–BA model was closer to the observed or reference point compared to the other models. Thus, it had better performance than the other models.
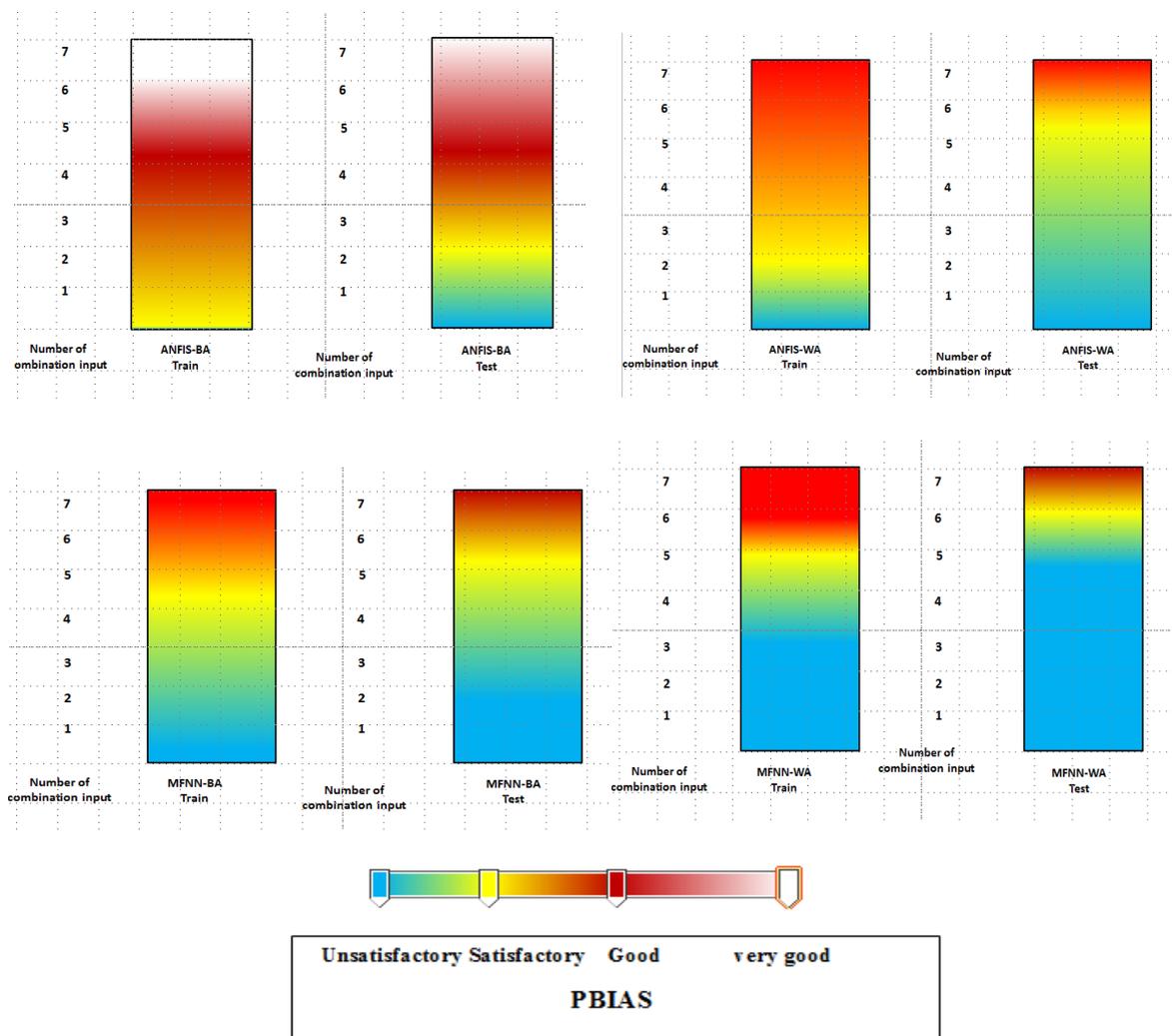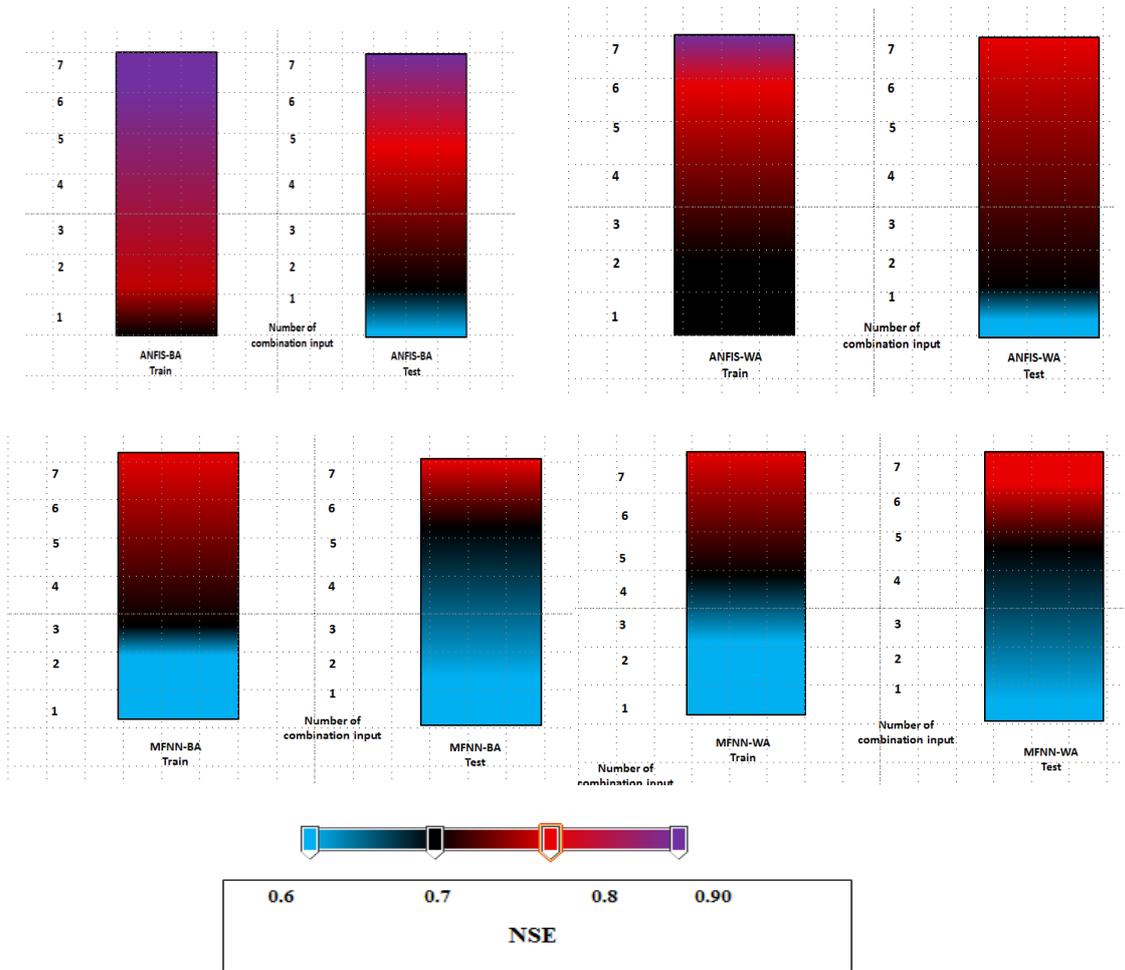

Figure 12. Computation of percent bias (PBIAS).

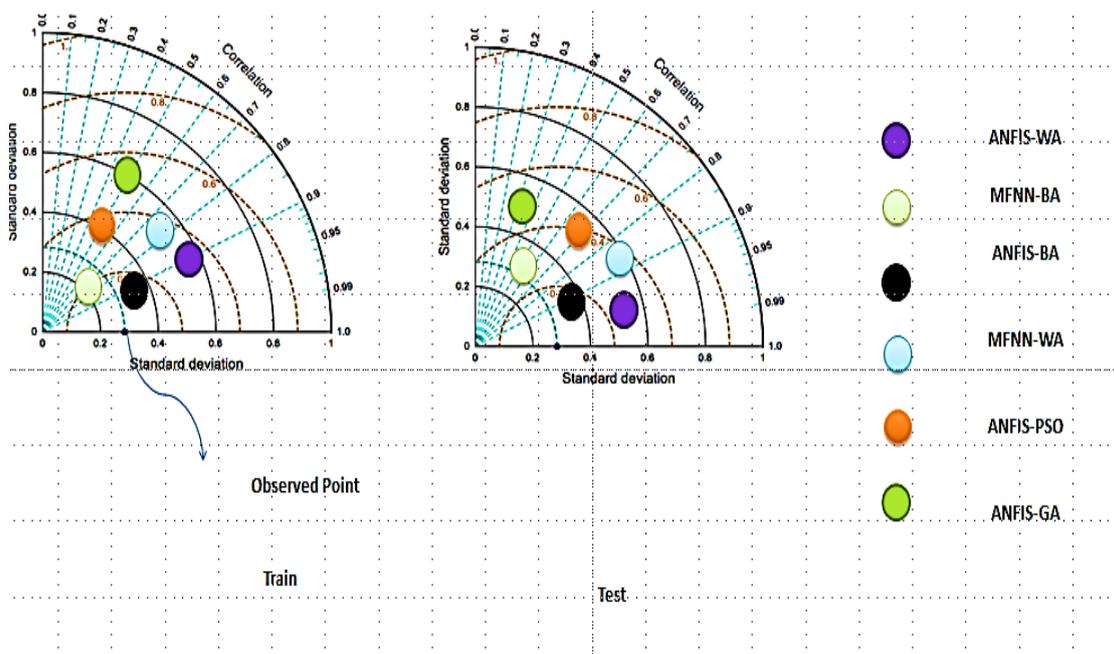**Figure 13.** Computation of Nash–Sutcliff efficiency (NSE).

Figure 14. Taylor diagram for the different models.

3.3. Discussion of Results

From the previous investigations, it can be noticed that the soft computing models are in the classification of box or semi box models. Thus, this results in complexity in the uncertainty of the study. In soft computing models, input data are applied to minimize the model's errors in the training (calibration) stage. In fact, the model's errors are dependent on the system's input data. Thus, it is important to examine the model's efficiency to determine the soft computing model's uncertainty generated by variations in initial data. In this article, a specified percentage of inputs was used to calibrate the soft computing models. The number of 2000 iterations was considered appropriate to carry out the input sampling cycle. For each random instance, 2000 model parameters were generated from calibrated patterns. In this method, uncertainty of all input data (e.g., precipitation, discharge, and SSL) were considered in the evaluation of model performances. Two important indices quantified the uncertainty values. The p index, percentage of measured data bracketed by 95% (95 PPU) uncertainty, was applied to measure the uncertainty values. First, the cumulative distribution was generated for the final outputs. The 95 PPU was computed at the lower (2.5%) and upper (97.5%) bounds of the obtained distribution. A p value of 1 showed that the model had a good performance. Average thickness of the 95 PPU was used to determine the r index. This index was computed by dividing the average thickness by the standard deviation of the measured data.

1. *p* index:

The extent of p indices varied from 0.6 to 0.90. As can be seen in Figure 11, the *p* values of 0.8 and 0.9 covered a large extent of the ANFIS–BA model. As observed in Figure 15, all models had the lowest *p* values for the first and second combination. The MFNN models had more uncertainty because the *p* values decreased with MFNN models. The MFNN–WA models had the worst performance because the *p* value varied from 0.6 to 0.7 to a large extent in these models.
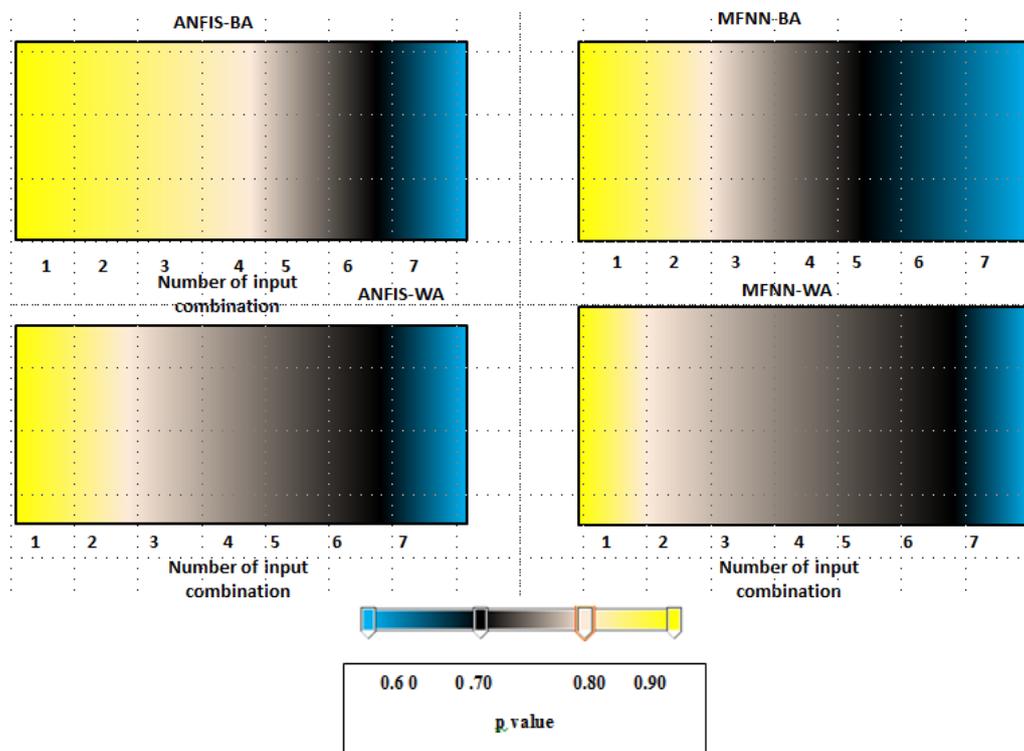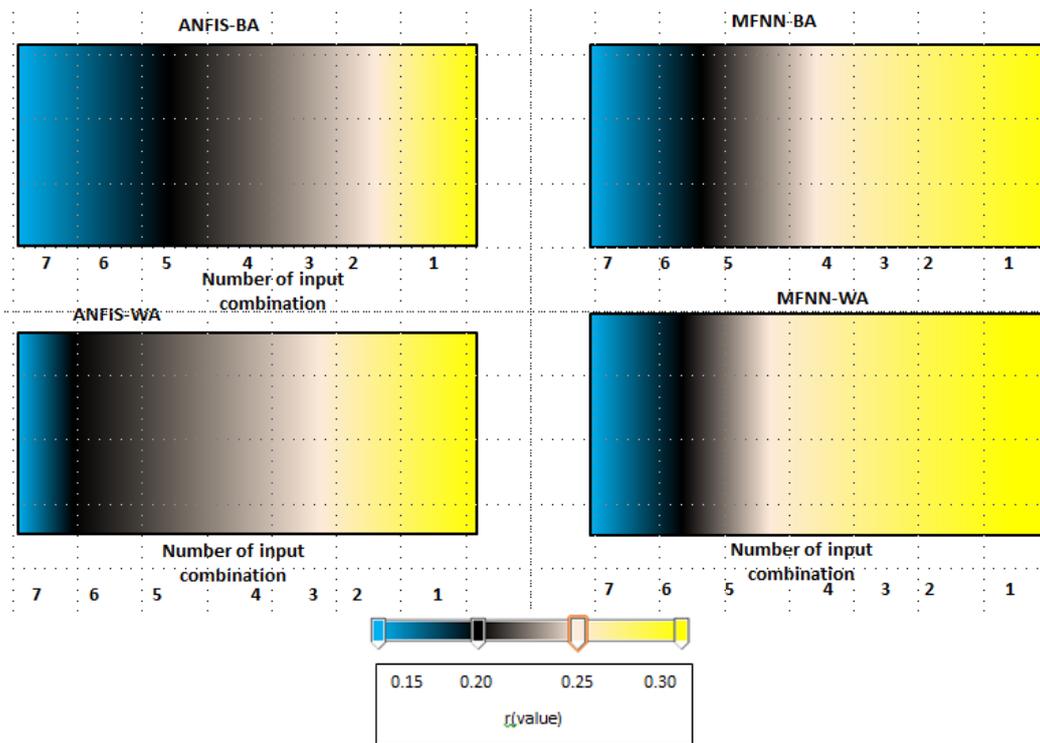

Figure 15. *Cont.*

Figure 15. Computation of uncertainty value.

2.　*r* indices

The average thickness of 95 PPU for the ANFIS–BA model was less than that of the other models because the *r* value varied from 0.15 to 0.20. A large extent of the MFNN–WA model had *r* values of 0.25 and 0.30 and thus had more uncertainty compared to the ANFIS–BA models. Generally, the previous results and uncertainty results showed that the ANFIS–BA and ANFIS–WA models had the best performances among all models.

Although different studies show that regression and ANN models have successful performances [35], this study showed that the ANFIS models were better than the ANN models. In addition, the results were different when different optimization algorithms were used to improve the ANFIS and ANN models. This study showed that dependence on the performance of the ANFIS and ANN models was high for SLL prediction. The general results indicated that the soft computing models with BA had better results than the prediction models with WA. The BA could accurately find the ANFIS and MFNN parameters. Thus, the selection of simple and accurate algorithms is important for soft computing models. The magnified maps were generated by an inverse distance weight (IDW) interpolation (Figure 16). Seven combinations were used for the soft computing models because the previous results showed that the combinations mentioned had the best performance among all models. The kappa index was used to evaluate the agreement between the simulated and observed maps. The improved ANFIS and MFNN models had better performances than the standalone ANFIS and MFNN models. In addition, the ANFIS–BA model generated more accurate results (kappa: 0.90) compared to the other models used.
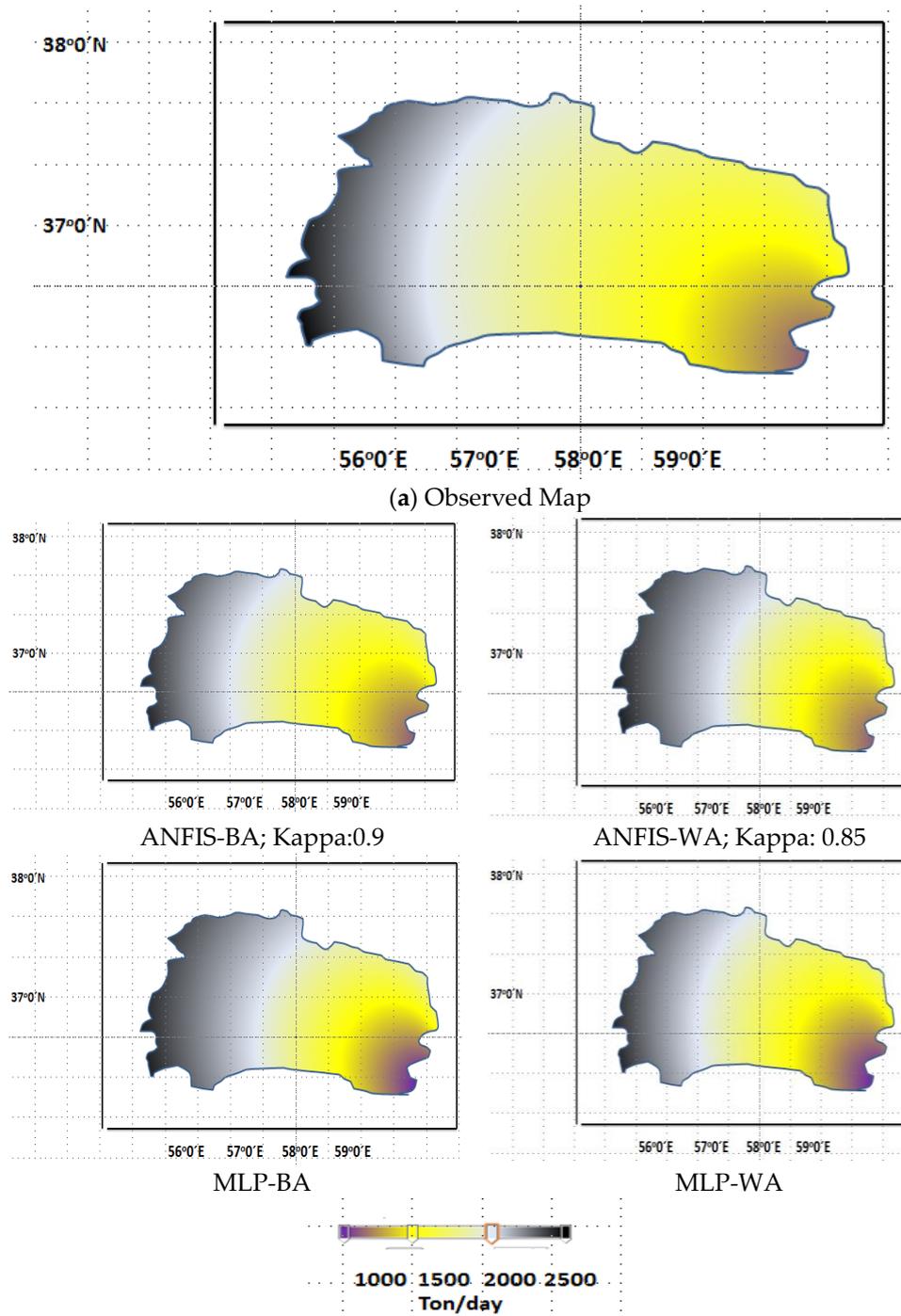
(**a**) Observed Map

ANFIS-BA; Kappa:0.9　　　　　　　　　　　ANFIS-WA; Kappa: 0.85

MLP-BA　　　　　　　　　　　　　MLP-WA

Figure 16. Monthly average suspended sediment load.

4. Conclusions

In this paper, the efficiencies of the soft computing models ANFIS–BA, ANFIS–WA, MFNN–BA, and MFNN–WA were examined for SSL prediction for the Atrek basin in Iran. The results were compared with single ANFIS and MFNN models and indicated that the ANFIS–BA and ANFIS–WA models had good performances for SSL prediction, being the best and the second-best ranked models, respectively. The ANFIS–BA model outperformed the other models based on statistical indices and uncertainty analysis. The findings of this article show good performances of BA and WA as optimization algorithms to enhance the efficiency of the ANFIS model for SSL prediction. Regarding the outputs,

future investigations can consider the performance of multiobjective algorithms for improving soft computing models. In addition, the current version of the BA and WA can be modified to have a better performance during the simulation process. The applied models had uncertainty in the simulation, and uncertainty computations showed that the ANFIS-BA had the best results with a lower uncertainty value. However, the evaluation of models can be completed when the models are evaluated in climate change conditions. In addition, the results of the models can be more comprehensive if the results are compared to the outputs of empirical models.

References

1. Liu, Q.J.; Zhang, H.Y.; Gao, K.T.; Xu, B.; Wu, J.Z.; Fang, N.F. Time-frequency analysis and simulation of the watershed suspended sediment concentration based on the Hilbert-Huang transform (HHT) and artificial neural network (ANN) methods: A case study in the Loess Plateau of China. *Catena* **2019**, *179*, 107–118. [CrossRef]

2. Sharghi, E.; Nourani, V.; Najafi, H.; Gokcekus, H. Conjunction of a newly proposed emotional ANN (EANN) and wavelet transform for suspended sediment load modeling. *Water Supply* **2019**, *19*, 1726–1734. [CrossRef]

3. Nourani, V.; Molajou, A.; Tajbakhsh, A.D.; Najafi, H. A Wavelet Based Data Mining Technique for Suspended Sediment Load Modeling. *Water Resour. Manag.* **2019**, *33*, 1769–1784. [CrossRef]

4. Abobakr Yahya, A.S.; Ahmed, A.N.; Binti Othman, F.; Ibrahim, R.K.; Afan, H.A.; El-Shafie, A.; Fai, C.M.; Hossain, M.S.; Ehteram, M.; Elshafie, A. Water Quality Prediction Model Based Support Vector Machine Model for Ungauged River Catchment under Dual Scenarios. *Water* **2019**, *11*, 1231. [CrossRef]

5. Tabatabaei, M.; Salehpour Jam, A.; Hosseini, S.A. Suspended sediment load prediction using non-dominated sorting genetic algorithm II. *Int. Soil Water Conserv. Res.* **2019**, *7*, 119–129. [CrossRef]

6. Hassanpour, F.; Sharifazari, S.; Ahmadaali, K.; Mohammadi, S.; Sheikhalipour, Z. Development of the FCM-SVR Hybrid Model for Estimating the Suspended Sediment Load. *KSCE J. Civ. Eng.* **2019**, *23*, 2514–2523. [CrossRef]

7. Rezakazemi, M.; Mosavi, A.; Shirazian, S. ANFIS pattern for molecular membranes separation optimization. *J. Mol. Liq.* **2019**, *274*, 470–476. [CrossRef]

8. Sihag, P.; Tiwari, N.K.; Ranjan, S. Prediction of unsaturated hydraulic conductivity using adaptive neuro-fuzzy inference system (ANFIS). *ISH J. Hydraul. Eng.* **2017**, *25*, 132–142. [CrossRef]

9. Karaboga, D.; Kaya, E. Training ANFIS by Using an Adaptive and Hybrid Artificial Bee Colony Algorithm (aABC) for the Identification of Nonlinear Static Systems. *Arab. J. Sci. Eng.* **2018**, *44*, 3531–3547. [CrossRef]

10. Ho, J.Y.; Afan, H.A.; El-Shafie, A.H.; Koting, S.B.; Mohd, N.S.; Jaafar, W.Z.B.; Lai Sai, H.; Malek, M.A.; Ahmed, A.N.; Mohtar, W.H.M.W.; et al. Towards a time and cost effective approach to water quality index class prediction. *J. Hydrol.* **2019**, *575*, 148–165. [CrossRef]

11. Gao, F.; Wang, Y.; Hu, X. Evaluation of the suitability of Landsat, MERIS, and MODIS for identifying spatial distribution patterns of total suspended matter from a self-organizing map (SOM) perspective. *Catena* **2019**, *172*, 699–710. [CrossRef]

12. Binns, A.D.; Fata, A.; Ferreira da Silva, A.M.; Bonakdari, H.; Gharabaghi, B. Modeling Performance of Sediment Control Wet Ponds at Two Construction Sites in Ontario, Canada. *J. Hydraul. Eng.* **2019**, *145*, 5019001. [CrossRef]

13. Guven, A.; Kişi, Ö. Estimation of Suspended Sediment Yield in Natural Rivers Using Machine-coded Linear Genetic Programming. *Water Resour. Manag.* **2010**, *25*, 691–704. [CrossRef]

14. Kisi, O.; Yaseen, Z.M. The potential of hybrid evolutionary fuzzy intelligence model for suspended sediment concentration prediction. *Catena* **2019**, *174*, 11–23. [CrossRef]

15. Choubin, B.; Darabi, H.; Rahmati, O.; Sajedi-Hosseini, F.; Kløve, B. River suspended sediment modelling using the CART model: A comparative study of machine learning techniques. *Sci. Total Environ.* **2018**, *615*, 272–281. [CrossRef] [PubMed]

16. Kakaei Lafdani, E.; Moghaddam Nia, A.; Ahmadi, A. Daily suspended sediment load prediction using artificial neural networks and support vector machines. *J. Hydrol.* **2013**, *478*, 50–62. [CrossRef]

17. Ebtehaj, I.; Bonakdari, H. Evaluation of Sediment Transport in Sewer using Artificial Neural Network. *Eng. Appl. Comput. Fluid Mech.* **2013**, *7*, 382–392. [CrossRef]

18. Melesse, A.M.; Ahmad, S.; McClain, M.E.; Wang, X.; Lim, Y.H. Suspended sediment load prediction of river systems: An artificial neural network approach. *Agric. Water Manag.* **2011**, *98*, 855–866. [CrossRef]

19. Shiau, J.-T.; Chen, T.-J. Quantile Regression-Based Probabilistic Estimation Scheme for Daily and Annual Suspended Sediment Loads. *Water Resour. Manag.* **2015**, *29*, 2805–2818. [CrossRef]

20. Chen, X.Y.; Chau, K.W. A Hybrid Double Feedforward Neural Network for Suspended Sediment Load Estimation. *Water Resour. Manag.* **2016**, *30*, 2179–2194. [CrossRef]

21. Shamaei, E.; Kaedi, M. Suspended sediment concentration estimation by stacking the genetic programming and neuro-fuzzy predictions. *Appl. Soft Comput.* **2016**, *45*, 187–196. [CrossRef]

22. Buyukyildiz, M.; Kumcu, S.Y. An Estimation of the Suspended Sediment Load Using Adaptive Network Based Fuzzy Inference System, Support Vector Machine and Artificial Neural Network Models. *Water Resour. Manag.* **2017**, *31*, 1343–1359. [CrossRef]

23. Mustafa, M.R.; Rezaur, R.B.; Saiedi, S.; Isa, M.H. River Suspended Sediment Prediction Using Various Multilayer Perceptron Neural Network Training Algorithms—A Case Study in Malaysia. *Water Resour. Manag.* **2012**, *26*, 1879–1897. [CrossRef]

24. Khosravi, K.; Mao, L.; Kisi, O.; Yaseen, Z.M.; Shahid, S. Quantifying hourly suspended sediment load using data mining models: Case study of a glacierized Andean catchment in Chile. *J. Hydrol.* **2018**, *567*, 165–179. [CrossRef]

25. Khan, M.Y.A.; Tian, F.; Hasan, F.; Chakrapani, G.J. Artificial neural network simulation for prediction of suspended sediment concentration in the River Ramganga, Ganges Basin, India. *Int. J. Sediment Res.* **2019**, *34*, 95–107. [CrossRef]

26. Cai, X.; Wang, H.; Cui, Z.; Cai, J.; Xue, Y.; Wang, L. Bat algorithm with triangle-flipping strategy for numerical optimization. *Int. J. Mach. Learn. Cybern.* **2017**, *9*, 199–215. [CrossRef]

27. Biswas, P.; Navaneethkrishnan, B.; Anand, G.; Omkar, S.N. System identification of a small scaled helicopter using simulated annealing algorithm. In Proceedings of the 2018 Indian Control Conference, Kanpur, India, 4–6 January 2018.

28. Peña-Angulo, D.; Nadal-Romero, E.; González-Hidalgo, J.C.; Albaladejo, J.; Andreu, V.; Bagarello, V.; Barhi, H.; Batalla, R.J.; Bernal, S.; Bienes, R.; et al. Spatial variability of the relationships of runoff and sediment yield with weather types throughout the Mediterranean basin. *J. Hydrol.* **2019**, *571*, 390–405. [CrossRef]

29. Ehteram, M.; Afan, H.A.; Dianatikhah, M.; Ahmed, A.N.; Fai, C.M.; Hossain, M.S.; Allawi, M.F.; Elshafie, A.; Ehteram, M.; Afan, H.A.; et al. Assessing the Predictability of an Improved ANFIS Model for Monthly Streamflow Using Lagged Climate Indices as Predictors. *Water* **2019**, *11*, 1130. [CrossRef]

30. Valikhan-Anaraki, M.; Mousavi, S.-F.; Farzin, S.; Karami, H.; Ehteram, M.; Kisi, O.; Chow, M.F.; Hossain, M.; Hayder, G.; Najah, A.-M.; et al. Development of a Novel Hybrid Optimization Algorithm for Minimizing Irrigation Deficiencies. *Sustainability* **2019**, *11*, 2337. [CrossRef]

31. Yaseen, Z.M.; Ehteram, M.; Sharafati, A.; Shahid, S.; Al-Ansari, N.; El-Shafie, A. The integration of nature-inspired algorithms with Least Square Support Vector regression models: Application to modeling river dissolved oxygen concentration. *Water* **2018**, *10*, 1124. [CrossRef]

32. Ehteram, M.; Ahmed, A.N.; Fai, C.M.; Afan, H.A.; El-Shafie, A. Accuracy Enhancement for Zone Mapping of a Solar Radiation Forecasting Based Multi-Objective Model for Better Management of the Generation of Renewable Energy. *Energies* **2019**, *12*, 2730. [CrossRef]

33. Tariq, Z.; Mahmoud, M.; Abdulraheem, A. Core log integration: A hybrid intelligent data-driven solution to improve elastic parameter prediction. *Neural Comput. Appl.* **2019**. [CrossRef]

34. Yilmaz, B.; Aras, E.; Nacar, S.; Kankal, M. Estimating suspended sediment load with multivariate adaptive regression spline, teaching-learning based optimization, and artificial bee colony models. *Sci. Total Environ.* **2018**, *639*, 826–840. [CrossRef] [PubMed]

35. Najah, A.-M.; El-Shafie, A.; Karim, O.A. An augmented Wavelet De-noising Technique with Neuro-Fuzzy Inference System for water quality prediction. *Int. J. Innov. Comput. Inf. Control* **2012**, *8*, 7055–7082.