



Prediction of daily suspended sediment load (SSL) using new optimization algorithms and soft computing models

Hamid Darabi¹ · Sedigheh Mohamadi² · Zahra Karimidastenaee¹ · Ozgur Kisi³ · Mohammad Ehteram⁴ · Ahmed ELShafie⁵ · Ali Torabi Haghighi¹

Accepted: 28 February 2021 / Published online: 26 March 2021
© The Author(s) 2021

Abstract

Accurate modeling and prediction of suspended sediment load (SSL) in rivers have an important role in environmental science and design of engineering structures and are vital for watershed management. Since different parameters such as rainfall, temperature, and discharge with the different lag times have significant effects on the SSL, quantifying and understanding nonlinear interactions of the sediment dynamics has always been a challenge. In this study, three soft computing models (multilayer perceptron (MLP), adaptive neuro-fuzzy system (ANFIS), and radial basis function neural network (RBFNN)) were used to predict daily SSL. Four optimization algorithms (sine–cosine algorithm (SCA), particle swarm optimization (PSO), firefly algorithm (FFA), and bat algorithm (BA)) were used to improve the capability of SSL prediction of the models. Data from gauging stations at the mouth of the Kasilian and Talar rivers in northern Iran were used in the analysis. The selection of input combinations for the models was based on principal component analysis (PCA). Uncertainty in sequential uncertainty fitting (SUFI-2) and performance indicators were used to assess the potential of models. Taylor diagrams were used to visualize the match between model output and observed values. Assessment of daily SSL predictions for Talar station revealed that ANFIS-SCA yielded the best results (RMSE (root mean square error): 934.2 ton/day, MAE (mean absolute error): 912.2 ton/day, NSE (Nash–Sutcliffe efficiency): 0.93, PBIAS: 0.12). ANFIS-SCA also yielded the best results for Kasilian station (RMSE: 1412.10 ton/day, MAE: 1403.4 ton/day, NSE: 0.92, PBIAS: 0.14). The Taylor diagram confirmed that ANFIS-SCA achieved the best match between observed and predicted values for various hydraulic and hydrological parameters at both Talar and Kasilian stations. Further, the models were tested in Egel Creek Basin, Indiana state, USA. The results indicated that the ANFIS-SCA model reduced RMSE by 15% and 21% compared to the MLP-SCA and RBFNN-SCA models in the training phase. Comparing models performance indicated that the ANFIS-SCA model could decrease MAE error compared to ANFIS-BA, ANFIS-PSO, ANFIS-FFA, and ANFIS models by 18%, 32%, 37%, and 49% in the training phase, respectively. The results indicated that the integration of optimization algorithms and soft computing models can improve the ability of models for predicting SSL. Additionally, the hybridization of soft computing models with optimization algorithms can decrease the uncertainty of models.

Keywords Bat algorithm · Particle swarm optimization · Firefly algorithm · Sediment prediction · Sine–cosine algorithm

1 Introduction

Sediment dynamics (transport and deposition) can cause environmental issues and concerns such as damage to aquatic ecosystems, declining quality of surface water and groundwater, and variations in reservoir recharge and river morphology (Afan et al. 2016; Shojaezadeh et al. 2018; Gholami et al. 2016; Guo et al. 2020; Ren et al. 2020).

Suspended sediment load (SSL) in watersheds is one of the most important hydraulic and hydrological parameters, which can impact the performance of hydraulic structures and water transfer projects. Additionally, sediments transported to reservoirs can reduce the reservoir capacity and affect operational policy, e.g., water supply, energy generation, and irrigation. Therefore, the estimation and prediction of SSL in rivers are vital tasks in the water resources management, and accurate results would help decision-making on river engineering, reservoir operation,

Extended author information available on the last page of the article

watershed management, and sustainable water resources (Yang et al. 2009; Downs et al. 2009; Akrami et al. 2013; Himanshu et al. 2017; Haghghi et al. 2019). Prediction of daily sediment can lead to optimal decisions for dam's outlet operation during the flood and conveying some part suspended sediment load to downstream area. Addressing short-term and long-term sediment dynamics is challenging owing to the heterogeneity of basins, the uncertainty in hydrological parameters, and the stochastic nature of flow and characteristics of sediment transport and deposition processes (Malmon et al. 2002; Pirmia et al. 2019b; Pizzuto 2020). Imprecise SSL modeling and prediction can reduce the amount of water stored by dam reservoirs, which can have an enormous negative impact on domestic and agricultural water supply, and also on dam structures (Lafdani et al. 2013; McCarney-Castle et al. 2017; Zhang et al. 2020; Zhao et al. 2020).

During recent decades, various approaches to improve the accuracy of SSL predictions have been introduced, including numerical and hydraulic, distributed and lumped models, statistics, empirical models, and machine learning models (Bezak et al. 2014; Merkhali et al. 2015; Kumar et al. 2016; Shamaei and Kaedi 2016; Choubin et al. 2018). Some studies have predicted SSL at daily scale using data-driven methods such as machine learning algorithms and soft computing models (Nourani and Andalib 2015; Choubin et al. 2018; Kaveh et al. 2020). Other studies worldwide seeking to enhance the precision of the SSL estimation have used machine learning techniques such as adaptive neuro-fuzzy system (ANFIS) (Rajaei et al. 2009; Cobaner et al. 2009; Kisi et al. 2012; Azamathulla et al. 2012; Vafakhah 2013; Choubin et al. 2018), artificial neural network (ANN) (Rajaei et al. 2009; Melesse et al. 2011; Kisi et al. 2012; Vafakhah 2013; Nourani and Andalib 2015; Wang et al. 2018; Halecki et al. 2018; Liu et al. 2019), support vector machine (SVM) (Kisi et al. 2012; Pektaş and Doğan 2015; Choubin et al. 2018), multilayer perceptron (MLP) (Cigizoglu 2004; Gholami et al. 2016; Romano et al. 2018), and radial basis function neural network (RBFNN) (Erol et al. 2008; Ahmad and Kumar 2016; Ibrahim et al. 2019). The soft computing models were widely applied for predicting SSL, e.g., Adib and Mahmoodi (2017) were applied GA method to optimize the structure of the ANN model predicting SSL, Talebi et al. (2017) estimated SSL using regression trees and ANN models, Salih et al. (2020) have illustrated that the attribute selected classifier performed better than the tree models in SSL prediction, Ehteram et al. (2020) have employed ANN and a multiobjective genetic algorithm to predict the SSL, and Samantary and Ghose (2020) estimated SSL using SVM, feed-forward neural network (FFN), and RBFNN and they have shown that the SVM had the highest performance.

Although the MLP, ANIFS, RBFNN, and SVM models have a high capability for estimating SSL, optimization of

these algorithms is required to obtain more accurate results (Fiyadh et al. 2019). Classical model training algorithms, such as backpropagation and the gradient descent algorithm, may become trapped in local optimums, so researchers have begun to develop new optimization algorithms (Ehteram et al. 2017). One recent example is the sine-cosine algorithm (SCA), inspired by mathematical sine and cosine functions, which has high search accuracy, speed of convergence, and stability (Mirjalili 2016). Optimization algorithms can be utilized as training algorithms to set the internal parameters of the MLP, ANFIS, and RBFNN models.

In the present study, data-based approaches and soft computing models (stand-alone and hybridized with optimization algorithms) were used for predicting SSL in the Talar river basin in northern Iran, where sediment is mostly generated during high-severity, erosive precipitation events and where complex processes determine suspended sediment and precipitation in river systems at watershed scale. The innovation of the present study is the new soft computing hybrid models which have been employed in previous studies for predicting other hydrological variables. Furthermore, the present study deals with using these soft computing models and optimization algorithms that can be linked to hydraulic and hydrological modeling. Additionally, the present study deals with the uncertainty of model parameters and its effect on the outcomes. ANFIS and ANN models are widely used models for predicting hydrological variables given their high potential, high accuracy, and easy learning for modelers. Furthermore, the extensive capability of soft computing models in other engineering fields makes the mentioned models be present as the models used in the study. However, the motivation behind of the study is to provide solutions to identify the parameters needed to estimate the SSL in different areas. Moreover, the optimization algorithms of the study, as will be stated, were selected for the study given their high search capability, fast convergence speed, and lack of computational complexity. The hybridization of the models makes the results more accurate. Likewise, the models are more able to simulate variables in more complex problems. Since the hybridized and optimized models performed better than individual models, the soft computing models are frequently optimized or hybridized to overwhelm the weakness of stand-alone models. However, it should be considered that the preparation of the structure of soft computing models and the selection of the best input scenario are the challenges of the current study.

The current study develops a low-cost estimation approach for accurately predicting SSL in developing regions where sediment loads in rivers are the main environmental concern. Specific objectives were to: (1) develop and implement optimization algorithms (SCA, particle

swarm optimization (PSO), bat algorithm (BA) and firefly algorithm (FFA) to improve model prediction of SSL; (2) investigate the capability of the four optimization methods in SSL prediction by applying widely used performance indices; and (3) compare outputs achieved using the stand-alone and hybrid ANFIS, MLP, and RBFNN models.

2 Material and methods

2.1 Study areas

This study was carried out in two case study including Talar river and Egel Creek Basins located in Iran and USA with different types of climate and environmental conditions. In the following section, some characteristics of the two above-mentioned basins are presented.

2.1.1 Talar river basin

The Talar river watershed (2100 km²) is situated in Mazandaran region, northern Iran (52° 35' 18"–53° 23' 35" E; 35° 44' 19"–36° 19' 13" N) (Fig. 1). Based on its aridity index of 0.69 (Sahin 2012; Pirnia et al. 2019a), the region climate is semi-humid, with 552.7 mm yearly precipitation and mean yearly minimum and maximum temperatures of 7.7 and 21.1 °C (Kavian et al. 2018). The smaller Kasilian river also runs through the watershed, to discharge into the Caspian Sea to the north (Fig. 1). Landslides are an important sediment source to both the Talar and Kasilian river systems (Emamgholizadeh and Demneh 2019). The watershed is characterized by intense rainfall events accompanied by frequent floods (Kavian et al. 2018) and has mountainous terrain characterized by rugged topography (altitude ranging from approximately 200 to 4000 m asl) and sparse vegetation cover in headwater areas, leading to huge sediment flows to the river network (Kavian et al. 2018). Both rivers have hydrometric stations situated at their outlet, from which daily observed data on rainfall, discharge, and suspended sediment concentration (SSC) were obtained for this work. The data were randomly divided into two subsets, with 80% utilized to calibrate the models and the remaining 20% utilized to test the proposed models. The maximum suspended sediment concentrations in the training and testing datasets were, respectively, 40,000 and 39,200 ton/day at Talar station and 60,000 and 59,000 ton/day at Kasilian station (Table 1).

2.1.2 Egel Creek Basin

In addition of the Talar basin, we used our models to predicts the daily SSL in a temperate and humid continental climate named Egel Creek Basin in the Indiana

state, USA (Fig. 2). The models run based on the rainfall, temperature, and discharge data (Table 1) from 2015 to 2018 (data retrieved from <https://www.usgs.gov/centers/oki-water>). For this basin, THE data were randomly divided into two subsets, with 80% utilized to calibrate the models and the remaining 20% utilized to test the proposed models.

2.2 Models tested for SSL prediction

2.2.1 Adaptive neuro-fuzzy system (ANFIS)

As an artificial neural network combined with fuzzy logical inference, ANFIS has a high ability for dealing with the imprecision and uncertainty of nonlinear environmental problems through its strong, effective learning techniques (Chang and Lai 2014; Choubin et al. 2018). Figure 3a shows a structure of the ANFIS model, which is a rule-based system comprising three parts: a rule base, a database, and an inference system that produces the system results by combining the fuzzy rules (Yurdusev and Firat 2009). The five layers in the ANFIS model are (1) input nodes, (2) rule nodes, (3) average nodes, (4) consequent nodes, and (5) output nodes, which employ different algorithms to produce fuzzy rules for training and testing (Park et al. 2012; Choubin et al. 2018). In ANFIS grid partitioning, fuzzy clustering and hybrid learning algorithms are applied to determine the input data structures in combination with the backpropagation gradient descent method (Cobaner et al. 2009; Kisi et al. 2012). The ANFIS model creates the following if–then rules using the pattern of input and output data:

$$\text{if}(x = A_1) \text{ and } (y = B_1) \rightarrow f = p_1x + q_1x + r_1 \quad (1)$$

$$\text{if}(x = A_2) \text{ and } (y = B_2) \rightarrow f = p_2x + q_2x + r_2 \quad (2)$$

where $A_1, B_1, A_2,$ and B_2 are related membership functions (MFs), x and y are inputs, and $p_1, q_1, r_1,$ and r_2 are consequent parameters. The ANFIS model has five computational layers:

1. The amount of input variable is fuzzified by the first layer:

$$O_{1,i} = \mu_{A_i}(x) \quad (3)$$

where $O_{1,i}$ is the MF of A_i and i is the linguistic label of node function. In the current work, the bell function was selected as the MF:

$$\mu_{A_i}(x) = \exp \left[- \left(\frac{x - c_i}{a_i} \right)^2 \right] \quad (4)$$

where a_i and c_i are premise parameters.

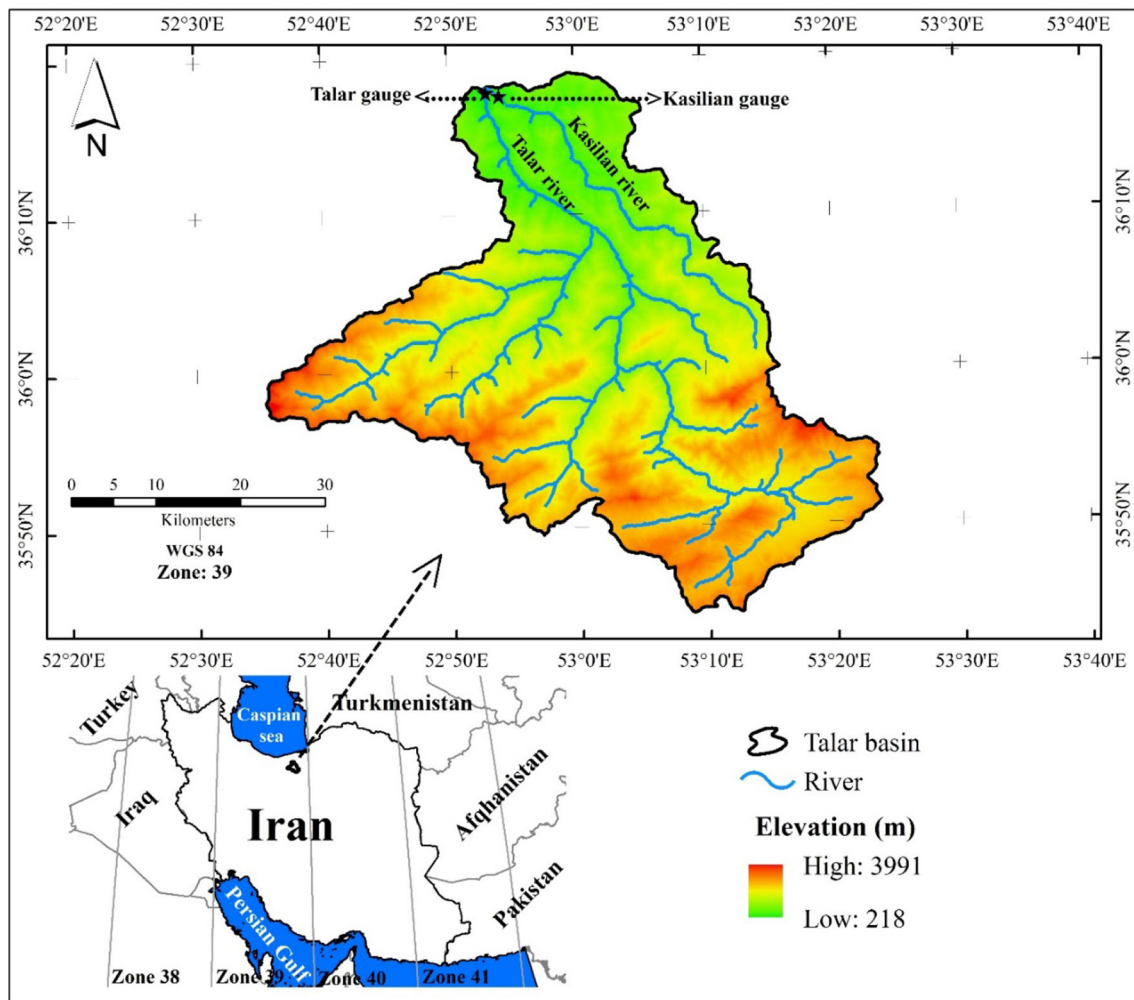


Fig. 1 Location of the Talar watershed, Iran

- The second layer calculates the firing strength of each rule by product operation:

$$O_{2,i} = \mu_{A_i}(x) \times \mu_{B_i}(x) \quad (5)$$

where $O_{2,i}$ is the second layer output and $\mu_{B_i}(x)$ is the fuzzy MF of fuzzy set B_i .

- The third layer is used to compute the normalized firing strength of every rule.

$$O_{3,i} = \frac{\omega_i}{\sum \omega_i} = \frac{\omega_i}{\omega_1 + \omega_2} \quad (6)$$

where ω_i is the fuzzy strength of each rule.

- The fourth layer determines the output of each rule:

$$O_{4,i} = \bar{\omega} \times f_i \quad (7)$$

where f_i is the output of the fuzzy region and $\bar{\omega}$ is the output of the third layer.

- The fifth layer is defuzzification:

$$O_5 = \sum \bar{\omega}_i \times f_i \quad (8)$$

where O_5 is the output of all the rules.

2.2.2 Multilayer perceptron (MLP)

The MLP network is a model one or more hidden layers which can use various input sets by a set of suitable outputs (Choubin et al. 2018). In MLP (Fig. 3b), the major learning rule is the backpropagation algorithm, which comprises two stages, a feed-forward and a backward stage, with external input information and calculated and measured information signals at the output (Cigizoglu 2004). The MLP network can simulate 90% of processes related to environmental and nature problems (Kim and Valdes 2003). The MLP model employed in the present study was a three-layer learning network having a hidden, an input, and an output layer (Samanta et al. 2019; Bhowmik et al. 2019; Van Dao et al. 2020). The neurons at hidden layers

Table 1 Characteristics and range of suspended sediment load (SSL) at gauging stations

For 2000–2011 period				
Parameter	Minimum	Maximum	Mean	SD
Training period (Talar)				
Discharge (Q , $m^3 s^{-1}$)	32	45	38	0.91
Rainfall (R , mm)	23	70	45	3.22
Sediment (ton/day)	545	40,000	34,567	747
Testing period (Talar)				
Discharge ($m^3 s^{-1}$)	35	45	39	0.92
Rainfall (mm)	25	72	48	3.98
Sediment (ton/day)	612	39,200	29,123	678
Training period (Kasilian)				
Discharge ($m^3 s^{-1}$)	33	67.00	42.00	1.12
Rainfall (mm)	29	8	4.00	3.85
Sediment (ton/day)	614	60,000	48,700	911
Testing period (Kasilian)				
Discharge ($m^3 s^{-1}$)	36	47	39	0.92
Rainfall(mm)	27	75	48	3.24
Sediment (ton/day)	616	59,000	34,567	905
For 2015–2018 period				
Parameter	Minimum	Maximum	Mean	SD
Training period (Eagel Creek Basin)				
Discharge (Q , $m^3 s^{-1}$)	0.06	163.25	51.89	12.14
Rainfall (R , mm)	20	90	65	14.55
Sediment (mg/L)	55	642	277.12	63.12
Testing period (Eagel Creek Basin)				
Discharge ($m^3 s^{-1}$)	1.12	154.12	7.89	14.15
Rainfall (mm)	45	80	67	12.98
Sediment (mg/L)	57	656	266.14	34.69

use the nonlinear activation function to provide the output as follows:

$$u_j = \sum_{i=1}^p w_{ij} \cdot x_i \tag{9}$$

$$x_j = \varphi(u_j - \theta_j) \tag{10}$$

where x_i is input, x_j is the output of the model, u_j is activation function, and θ_j is a threshold function. Previous researchers have successfully utilized the logistic sigmoid function for the MLP model as follows:

$$\varphi_x = \frac{1}{1 + e^{-x}} \tag{11}$$

The training algorithms are introduced to search for the optimum value of weight connections. Classical training algorithms such as backpropagation algorithm and gradient descent algorithm are widely applied to calibrate the MLP parameters.

2.2.3 Radial basis function neural network (RBFNN)

The RBFNN model is a type of feed-forward neural network which consists of a number of artificial neurons (see Fig. 3c). It can be considered a general-purpose network that can be employed in different fields to achieve accurate predictions. The RBFNN is considered a good candidate for solving problems by faster learning potential (Erol et al. 2008; Han et al. 2012; Kong et al. 2016; Ibrahim et al. 2019). RBFNN has very powerful mathematical functions for organization of deep learning theory in solving problems (Sabour and Movahed 2017). In practical application, the learning algorithm for the RBFNN model employs different datasets for training and testing, so as to adapt itself rapidly to new factors or combinations (Sabour and Movahed 2017). RBFNN has the advantage over other types of neural networks of having a clustering stage in training and testing (Singh et al. 2014; Kumar et al. 2016). It uses symmetric basis functions as activation functions:

$$\phi_i(x) = \exp\left(-\frac{\|x - c_i\|}{\sigma_i^2}\right) \tag{12}$$

where $\phi_i(x)$ is the Gaussian function, σ_i is the width of the i th radial basis function node, and c_i is the center of hidden neuron i . The network output is computed as follows:

$$y = \sum_{i=0}^n \phi_i w_i \tag{13}$$

where y is output and n is number of hidden neurons.

The training algorithms are used to set the RBFNN parameters such as center, width, and weight of the radial basis function node.

2.3 Optimization algorithms tested

2.3.1 Sine-cosine algorithm (SCA)

The SCA approach was first proposed by Mirjalili (2016). It updates the position of solutions using sine and cosine functions. The mathematical formulation of SCA is:

$$X_i^{t+1} = X_i^t + r_1 \times (\sin(r_2)) \times |r_3 p_i^t - x_i^t| \tag{14}$$

$$X_i^{t+1} = X_i^t + r_1 \times (\cos(r_2)) \times |r_3 p_i^t - x_i^t| \tag{15}$$

where X_i^t is the position of current solution at the i th iteration in the i th dimension, r_2 and r_3 are random values, p_i^t

Fig. 2 Location of Egel Creek Basin, USA

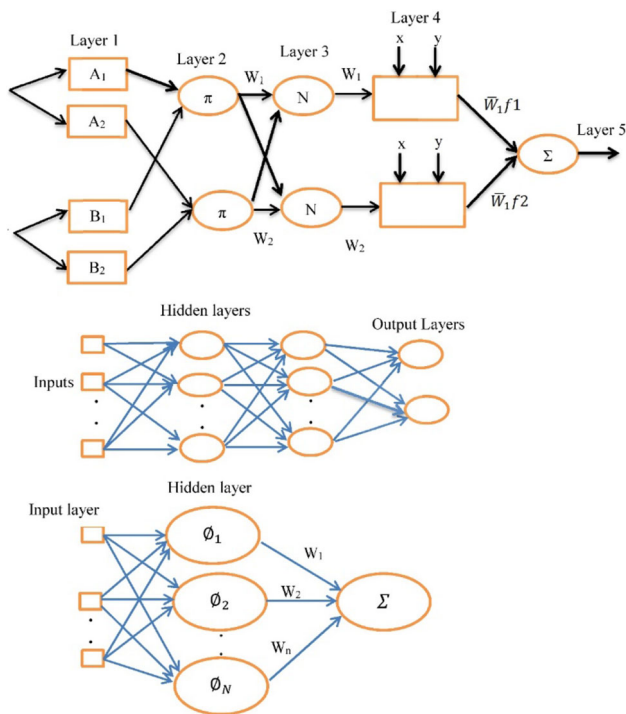
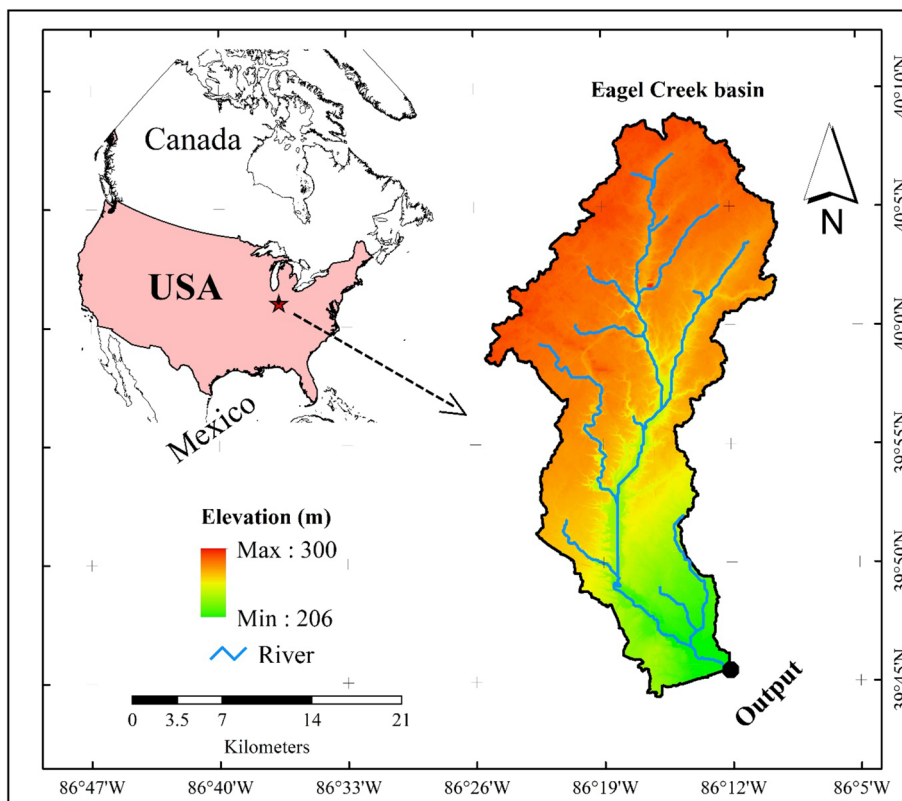


Fig. 3 Structure of the three models used: **a** adaptive neuro-fuzzy system (ANFIS), **b** multilayer perceptron (MLP), and **c** radial basis function neural network (RBFNN)

is the destination solution, and r_1 is a control parameter used to get a balance between exploration and exploitation. The two SCA functions (Eqs. 14, 15) are then integrated into one function:

$$X_i^t = \begin{cases} X_i^{t+1} = X_i^t + r_1 \times (\sin(r_2)) \times |r_3 p_i^t - x_i^t| & (\text{if } r_4 < 0.5) \\ X_i^{t+1} = X_i^t + r_1 \times (\cos(r_2)) \times |r_3 p_i^t - x_i^t| & (\text{if } r_4 \geq 0.5) \end{cases} \quad (16)$$

The following equation is used to update the value of parameter r_1 :

$$r_1 = a - t \frac{a}{T} \quad (17)$$

where a is a constant and T is the maximum quantity of iterations. Parameter r_2 is utilized to obtain the movement direction of the next solution. Parameter r_3 is used to define a random weight for the destination with a stochastic influence emphasizing ($r_3 > 1$) or decreasing distance ($r_3 < 1$). Parameter r_4 is used to switch between the cosine and sine functions. Figure 4 shows the sine–cosine effect on the next position and Fig. 5 shows a flowchart of SCA.

2.3.2 Bat algorithm (BA)

All bats have the echolocation characteristic to sense distance and use it to identify the difference between the food and obstacles (Yang et al. 2009). In the first step in BA, the

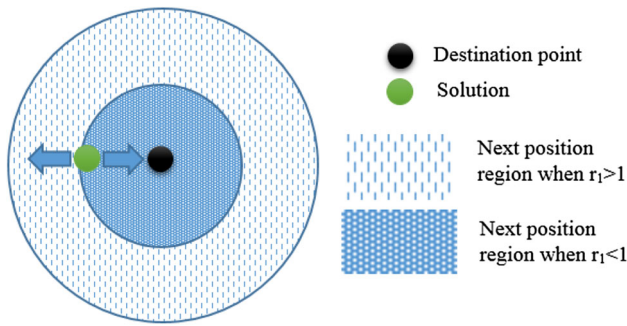


Fig. 4 Sine-cosine effect on the next position in sine-cosine algorithm (SCA)

initial population of bats is randomly initialized (Fig. 6). The BA uses the following equations to renew the bats' velocity and position (Yang et al. 2009):

$$\begin{aligned}
 f_i &= f_{\min} + (f_{\max} - f_{\min})\beta \\
 v_i^t &= v_i^{t-1} + (x_i^{t-1} - x^*)f_i \\
 x_i &= x_i^{t-1} + v_i^t
 \end{aligned}
 \tag{18}$$

where f_i is frequency of bat i , f_{\max} is the maximum frequency, f_{\min} is the minimum frequency, v_i^t is velocity of agent i at iteration t , x_i^{t-1} is position of agent i at iteration $t - 1$, x_i^t is position of agent i at iteration t , β is a random number, v_i^{t-1} is velocity of agent i at iteration t , x^* is the best solution, and x_i is position of agent i at iteration t . When the bat becomes closer to its prey, the rate of pulse emission and the loudness of the bat are renewed as:

$$\begin{aligned}
 A_i^{t+1} &= \alpha A_i^t \\
 r_i^{t+1} &= r_i^o [1 - \exp(-\gamma t)]
 \end{aligned}
 \tag{19}$$

where A_i^{t+1} is loudness of bat i at iteration $t + 1$, α is a constant value, γ is a constant value, r_i^o is the pulse emission' initial rate, r_i^{t+1} is the pulse emission rate of bat i at iteration $t + 1$, and A_i^t is the loudness of bat i at iteration t .

The bats use random walk to update their position:

$$x_{new} = x_{old} + \varepsilon A_i^t
 \tag{20}$$

where x_{new} is the bat's new position, x_{old} is the old position of the bat, and ε is a random number.

2.3.3 Firefly algorithm (FFA)

Firefly algorithm, introduced by Yang et al. (2009), is dependent on the firefly's behavior (Fig. 7). The short, rhythmic flashes produced by fireflies are intended to attract other fireflies that have weaker flashes. The landscape of the objective function identifies the firefly brightness. For a problem of minimization, a brighter firefly has a smaller objective function. The fireflies update their position as follows:

$$x_i(t + 1) = x_i(t) + \chi(r)(x_j(t) - x_i(t)) + \phi_t v
 \tag{21}$$

where $x_i(t + 1)$ is position of firefly i at iteration $t + 1$, $x_i(t)$ is position of firefly i at iteration t , $x_j(t)$ is position of firefly j at iteration t , $\chi(r)$ is attractiveness, ϕ_t is a step factor, and v is a random number. The attractiveness is computed as follows:

$$\chi(r) = \chi_0 e^{-\gamma r^2}
 \tag{22}$$

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{i=1}^D (x_{id} - x_{jd})^2}
 \tag{23}$$

Fig. 5 Flowchart of optimization using sine-cosine algorithm (SCA)

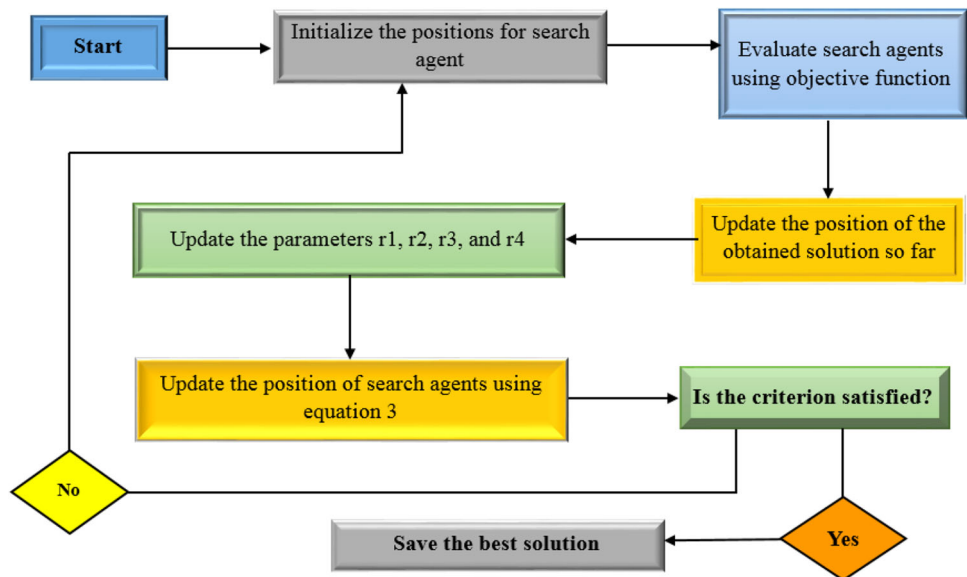


Fig. 6 Flowchart of optimization using bat algorithm (BA)

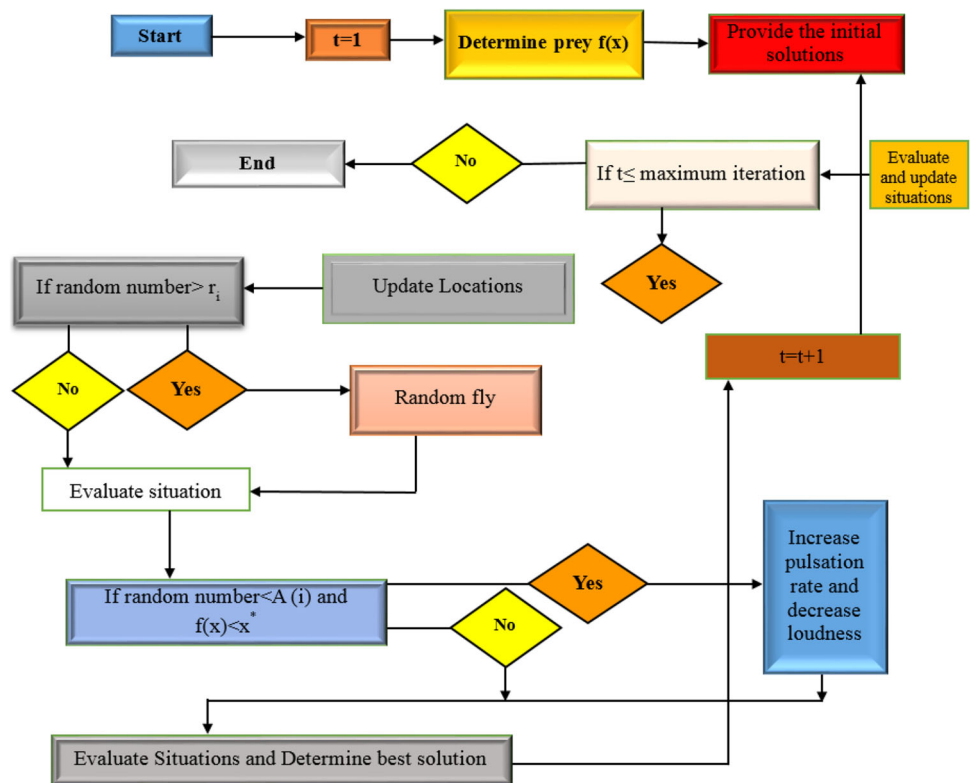
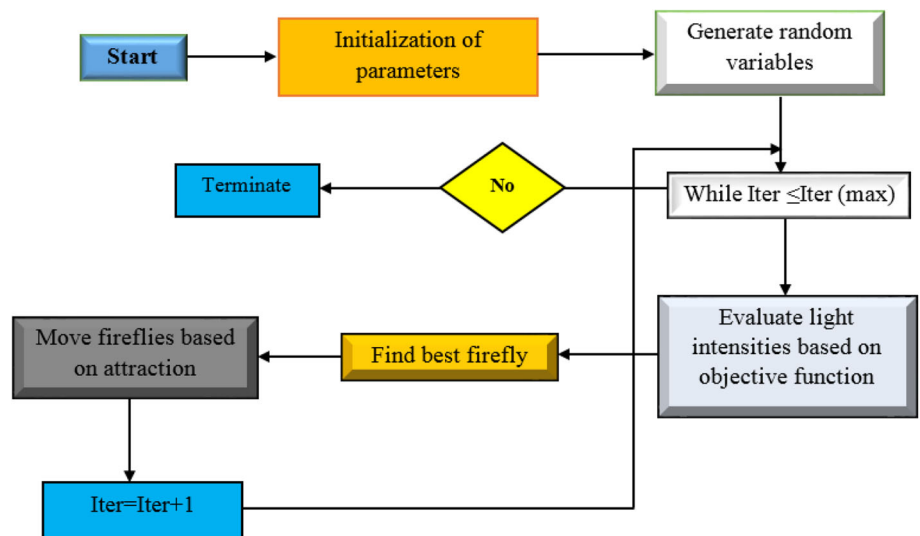


Fig. 7 Flowchart of optimization using firefly algorithm (FFA)



where χ_0 is attractiveness at $r = 0$, D is number of dimensions, and r_{ij} is distance between two fireflies.

2.3.4 Particle swarm optimization (PSO)

Eberhart and Kennedy (1995) introduced PSO, which is inspired by the social behavior of particles (Fig. 8). The algorithm starts with initialization of random particles in the search space. The particles search for the optimal

solution by updating generations. At each iteration, the two best values are used to update each particle. The first is the best solution found so far and the second is the best value found so far by any particle in the population. The following equations are utilized to renew the position and velocity of particles:

$$x_i(t) = x_i(t-1) + v_i(t) \quad (24)$$

$$v_i^t = wv_i^{t-1} + C_1r_1[P_{\text{best}} - x_i(t)] + C_2r_2[G_{\text{best}} - x_i^t] \quad (25)$$

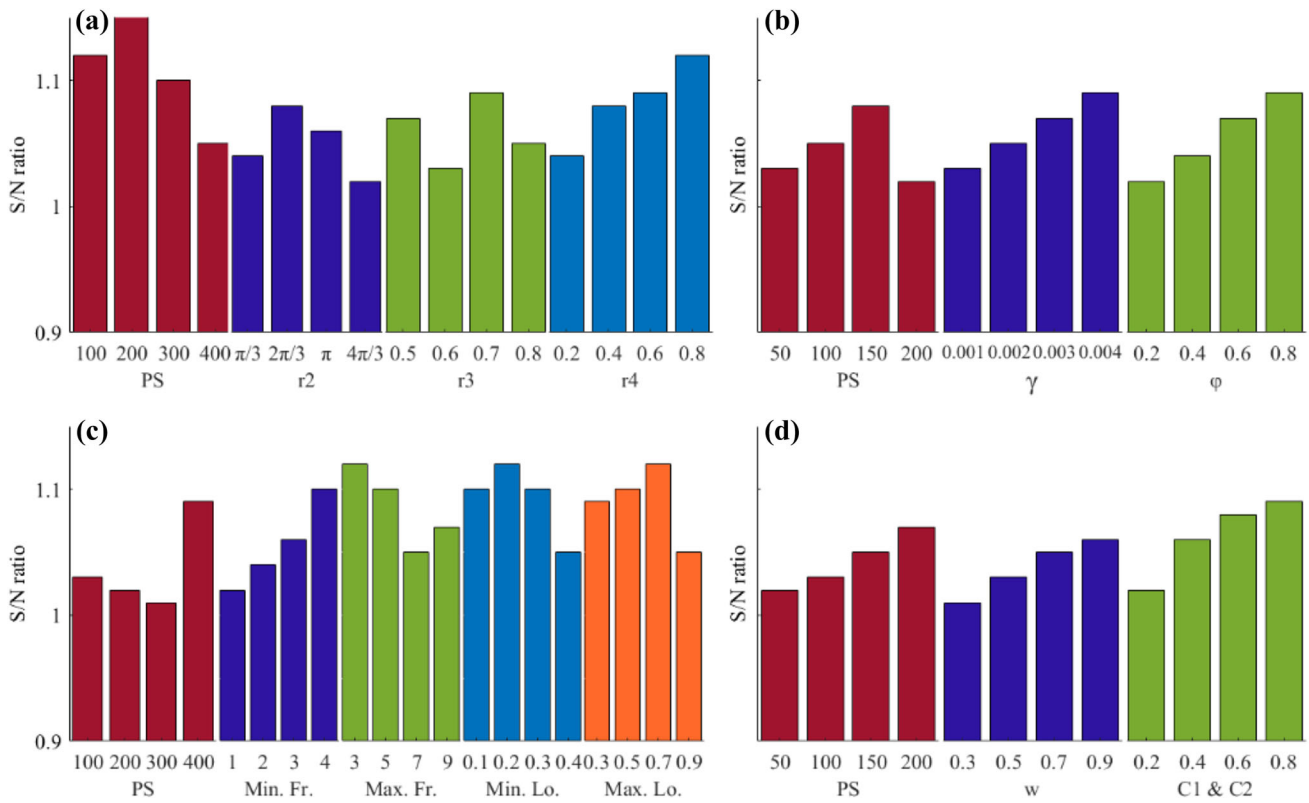


Fig. 8 Optimal parameters of optimization algorithms based on signal-to-noise (*S/N*) ratio. **a** Sine –cosine algorithm (SCA), **b** firefly algorithm (FFA), **c** bat algorithm (BA), and **d** particle swarm

optimization (PSO). *PS* population size, *Lo* loudness, *Fr* frequency. For explanation of parameters, see Sect. 2.2

where v_i^t is velocity of the particle at time t , w is inertia coefficient, C_1 and C_2 are acceleration coefficients, r_1 and r_2 are random numbers, P_{best} is the most promising position of the particle, G_{best} is the most promising position among the particles of the whole swarm, and $x_i(t)$ is the position of particles at time t .

2.4 Hybridization of prediction models with optimization algorithms

2.4.1 ANFIS hybridization

Application of the ANFIS model starts with setting parameters to optimal values, commonly by using a hybrid learning method combining gradient descent (GD) and the least square estimate (LSE). However, the hybrid LSE-GE method may unable to achieve the rate of convergence for finding appropriate values of internal parameters in ANFIS, and therefore supporting algorithms are widely applied to optimize the internal parameters. The premise and consequent parameters in ANFIS are decision variables of the optimization algorithms that are optimized using these supporting algorithms. The main function of the optimization algorithms is then to update the initial values of

the internal parameters in ANFIS, utilizing algorithm operators. An objective function, root mean square error (RMSE), is defined for hybrid ANFIS optimization algorithms. The optimization process tries to minimize the value of RMSE. When the ANFIS optimization algorithms converge to the lowest value of RMSE as a stopping criterion, the hybrid ANFIS model achieves the optimal value of its internal parameters.

2.4.2 MLP hybridization

The MLP parameters must be optimized to achieve the most accurate results. Training algorithms are required to set weight connections and threshold values. The initial threshold values and weight connections are defined as the initial population of algorithms. Each of the agents of the algorithms has two key parts: a set of weight connections and a set of threshold values. The values of MLP parameters are updated when the optimization algorithm tries to minimize the error function (RMSE). The convergence cycle of optimization continues until the hybrid MLP optimization algorithm model converges to a minimum objective function value.

2.4.3 RBFNN hybridization

Training algorithms are introduced to search for optimum parameters of the RBFNN model. Each of the agents of optimization algorithms has three key parts: center, width, and weight of the radial basis function node. The RBFNN parameters are defined as the initial population algorithms, which are entered into optimization algorithms to be updated by the operators of optimization algorithms. The optimal value of RBFNN parameters is found when the hybrid RBFNN optimization algorithm model converges to the lowest value of the target objective function.

2.5 Uncertainty analysis of soft computing models

Uncertainty in sequential uncertainty fitting (SUFI-2) is one of the best-known models for uncertainty analysis (Kumar et al. 2017). In SUFI-2, the parameter uncertainties account for uncertainty in model inputs and an objective function must be defined before the uncertainty analysis. Latin hypercube (LH) sampling is conducted, the objective function is assessed, and finally, the parameter covariance matrix is computed. In addition, 95% prediction uncertainty (95PPU) is computed at the 2.5% and 97% levels. Uncertainty analysis is required for the study as optimization algorithms try to find the exact values of the model parameters, as the input values may have include some sort of uncertainties. Thus, model uncertainty analysis can examine the effect of uncertainty related to model structure and parameter on the results. Two indices are used to quantify the uncertainty of models, observed data's percentage bracketed by 95 PPU (p index) and an index r computed as follows:

$$\bar{d} = \frac{1}{n} \sum_{t=1}^n (y_{97.5\%} - y_{2.5\%}) \quad (26)$$

$$r = \frac{\bar{d}}{\sigma}$$

where σ is standard deviation of the data, $y_{97.5\%}$ is the upper boundary of 95PPU, $y_{2.5\%}$ is the lower boundary of 95PPU, n is quantity of data, and r is average width of the confidence interval band. Other evaluation statistics utilized in this study were: RMSE (lower RMSE shows more accurate estimations), mean absolute error (MAE) (lower MAE shows more accurate estimation), percentage bias (PBIAS) (lower PBIAS shows more accurate estimations), and Nash–Sutcliffe efficiency (NSE) (NSE = 1 shows the ideal model):

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_{\text{obs}} - Y_{\text{sim}})^2} \quad (27)$$

$$\text{RMSE} = \frac{1}{n} \sum_{i=1}^n |Y_{\text{obs}} - Y_{\text{sim}}| \quad (28)$$

$$\text{PBIAS} = 100 * \left[\frac{\sum_{i=1}^n (Y_{\text{obs}} - Y_{\text{sim}})}{\sum_{i=1}^n (Y_{\text{obs}})} \right] \quad (29)$$

$$\text{NSE} = 1 - \left[\frac{\sum_{i=1}^n (Y_{\text{obs}} - Y_{\text{sim}})^2}{\sum_{i=1}^n (Y_{\text{obs}} - \bar{Y}_{\text{obs}})^2} \right] \quad (30)$$

where n is quantity of observed data, Y_{obs} is observed data, Y_{sim} is simulated data, and \bar{Y}_{obs} is mean of observed data.

3 Results and discussion

3.1 Selection of appropriate inputs for soft computing models

In this study, the soft computing models are used to predict SSL (t) (a 1-day ahead forecast of SSL). Principal component analysis (PCA) is an effective method for identifying inputs of models and decreasing the number of input parameters required (Lu et al. 2019). PCA achieves parsimony by describing the maximum value of common variance in a correlation matrix using the smallest number of illustrative concepts. The Kaiser–Meyer–Melkin criterion (KMO) is used to investigate the adequacy of data as follows. The KMO is a measure of the proportion of variance among variables that might be common variance (Darabi et al. 2014).

$$\text{KMO} = \frac{\sum (\text{correlation})^2}{\sum (\text{correlation})^2 + \sum (\text{partial correlation})^2} \quad (31)$$

According to the literature, the minimum value of KMO should be 0.5. In this study, KMO was 0.65. The correlation among variables should be checked, to avoid multicollinearity problems (Lu et al. 2019). In this study, all correlation values were below the threshold (0.9), and thus, there were no problems of multicollinearity. Table 2 shows the value of the contribution of principal components (PCs). The results indicated that the first three PCs included 60, 23, and 12% of input variables at Talar station, and 61, 20, and 11% of input variables at Kasilian station. Lagged data (one-day to nine-day lagged rainfall, one-day to nine-day lagged discharge, and one-day to nine-day lagged SSL) were regarded as the initial data. It was found that the first three PCs were affected more by one-day and two-day lagged SSL, one-day lagged R , and one-day lagged Q than by any other variables (Table 2). The direction of new future space was determined by the eigenvectors and the

Table 2 Eigenvalues and eigenvectors obtained for principal components (PCs) describing the most influential variables (time (t)-lagged suspended sediment load (SSL), discharge (Q), rainfall (R))

Parameter	Talar								
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
SSL (t - 1)	0.92	0.88	0.80	0.65	0.45	0.32	0.22	0.20	0.15
Q (t - 1)	0.87	0.85	0.82	0.79	0.77	0.72	0.68	0.67	0.66
R (t - 1)	0.84	0.83	0.81	0.80	0.78	0.76	0.64	0.62	0.60
SSL (t - 2)	0.83	0.82	0.80	0.78	0.65	0.62	0.60	0.57	0.55
SSL (t - 3)	0.79	0.77	0.74	0.73	0.69	0.68	0.57	0.56	0.54
Q (t - 2)	0.76	0.65	0.62	0.60	0.59	0.66	0.61	0.60	0.52
Q (t - 3)	0.67	0.60	0.60	0.58	0.55	0.65	0.60	0.58	0.54
R (t - 2)	0.53	0.45	0.41	0.39	0.37	0.30	0.29	0.27	0.20
R (t - 3)	0.45	0.33	0.29	0.27	0.26	0.25	0.27	0.25	0.19
Eigenvalue	5.45	2.10	1.10	0.20	0.09	0.03	0.02	0.05	0.05
Parameter	Kasilian								
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
SSL (t - 1)	0.93	0.89	0.80	0.67	0.43	0.31	0.21	0.19	0.14
Q (t - 1)	0.89	0.86	0.84	0.75	0.76	0.70	0.67	0.65	0.65
R (t - 1)	0.87	0.84	0.83	0.79	0.77	0.75	0.62	0.60	0.58
SSL (t - 2)	0.84	0.85	0.80	0.77	0.62	0.65	0.58	0.55	0.54
SSL (t - 3)	0.78	0.79	0.73	0.72	0.67	0.67	0.55	0.55	0.52
Q (t - 2)	0.75	0.69	0.64	0.61	0.54	0.65	0.60	0.58	0.51
Q (t - 3)	0.65	0.62	0.62	0.57	0.52	0.64	0.55	0.57	0.50
R (t - 2)	0.52	0.47	0.40	0.37	0.36	0.29	0.27	0.26	0.20
R (t - 3)	0.44	0.39	0.30	0.25	0.25	0.21	0.25	0.25	0.19
Eigenvalue	5.55	2.00	1.00	0.25	0.10	0.04	0.03	0.015	0.015
Parameter	Eagel Creek Basin								
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
SSL (t - 1)	0.93	0.90	0.86	0.84	0.82	0.78	0.65	0.62	0.55
Q (t - 1)	0.90	0.89	0.85	0.82	0.80	0.76	0.64	0.59	0.54
R (t - 1)	0.89	0.85	0.84	0.80	0.79	0.74	0.72	0.68	0.53
SSL (t - 2)	0.69	0.68	0.67	0.66	0.64	0.62	0.60	0.59	0.58
Q (t - 3)	0.65	0.62	0.60	0.59	0.54	0.52	0.50	0.45	0.42
SSL (t - 3)	0.64	0.55	0.54	0.52	0.50	0.49	0.44	0.43	0.41
Q (t - 3)	0.62	0.45	0.42	0.40	0.39	0.37	0.37	0.35	0.33
R (t - 2)	0.61	0.44	0.40	0.39	0.38	0.35	0.32	0.31	0.29
R (t - 3)	0.55	0.42	0.39	0.35	0.34	0.32	0.28	0.27	0.27
Eigenvalue	6.12	1.22	1.14	0.19	0.15	0.06	0.04	0.04	0.04

variance of data by the eigenvalues (Table 2). The PCs are the integration of the independent variable.

3.1.1 Tuning the random parameters in optimization algorithms

In the current work, the Taguchi model was utilized to set the random parameters of evolutionary algorithms. Population size and r_2 , r_3 , and r_4 are regarded as the random parameters in SCA that can affect the accuracy of the

proposed model. Four levels were defined for each of these four parameters. The total number of tests to be performed to find the optimum value of parameters was computed as:

$$\text{Number} = 1 + N(L - 1) \tag{32}$$

where L is level number and N is parameter number. Hence, at least 13 experiments had to be conducted for SCA. In addition, the Taguchi model utilizes signal-to-noise ratio to select the optimal value of parameters (Mozdgir et al. 2013):

Table 3 Value of evaluation criteria (root mean square error (RMSE), mean absolute error (MAE), Nash–Sutcliffe efficiency (NSE), percentage bias (PBIAS)) obtained for the different hybrid and stand-alone models in the training and testing phases for the Talar and Kasilian stations

Model	Train							
	RMSE (ton/day)		MAE (ton/day)		NSE		PBIAS	
	Talar	Kasilian	Talar	Kasilian	Talar	Kasilian	Talar	Kasilian
ANFIS-SCA	934.2	898.1	912.2	712.3	0.93	0.95	0.14	0.12
MLP-SCA	967.1	923.2	958.3	823.10	0.92	0.93	0.18	0.14
RBFNN-SCA	1056.3	1012.1	1043.2	1000.2	0.90	0.91	0.20	0.16
ANFIS-BA	1012.4	923.2	1001.1	912.32	0.89	0.94	0.22	0.18
MLP-BA	1102.5	1098.1	1076.2	967.21	0.87	0.92	0.23	0.20
RBFNN-BA	1145.7	1255.2	1112.4	1123.2	0.86	0.90	0.25	0.23
ANIFS-PSO	1212.2	1044.1	1204.3	1000.1	0.85	0.93	0.27	0.25
MLP-PSO	1289.2	1245.6	1232.3	1189.2	0.84	0.90	0.29	0.21
RBFNN-PSO	13,450	1469.2	1268.2	1456.8	0.83	0.89	0.31	0.26
ANFIS-FFA	1698.2	1589.6	1598.9	1512.30	0.82	0.88	0.32	0.28
MLP-FFA	1701.2	1590.20	1685.2	1578.3	0.81	0.87	0.32	0.30
RBFNN-FFA	1754.1	1592.3	1697.2	1589.2	0.80	0.85	0.33	0.31
ANFIS	1759.8	1623.3	1699.2	1612.3	0.79	0.84	0.34	0.33
MLP	1772.3	1645.2	1701.3	1639.2	0.78	0.83	0.35	0.35
RBFNN	1789.10	1655.6	1765.2	1645.2	0.77	0.82	0.36	0.39

Model	Test							
	Talar	Kasilian	Talar	Kasilian	Talar	Kasilian	Talar	Kasilian
ANFIS-SCA	1423.2	1412.10	1412.2	1403.4	0.92	0.92	0.16	0.14
MLP-SCA	1455.7	1423.2	1423.8	1421.2	0.91	0.90	0.20	0.17
RBFNN-SCA	1497.6	1467.2	1478.6	1455.2	0.89	0.89	0.24	0.18
ANFIS-BA	1434.2	1422.2	1423.4	1418.2	0.88	0.90	0.27	0.21
MLP-BA	1502.3	1501.1	1498.4	1498.2	0.86	0.86	0.28	0.22
RBFNN-BA	1595.1	1578.2	1512.3	1500.2	0.85	0.84	0.29	0.24
ANIFS-PSO	1602.3	1601.2	1601.2	1598.2	0.84	0.78	0.32	0.27
MLP-PSO	1665.8	1662.3	1663.1	1661.2	0.83	0.76	0.33	0.29
RBFNN-PSO	1678.2	1675.2	1672.3	1651.2	0.81	0.75	0.35	0.32
ANFIS-FFA	1692.9	1687.2	1678.1	1678.0	0.80	0.73	0.36	0.34
MLP-FFA	1723.8	1712.3	1712.9	1711.2	0.79	0.71	0.38	0.39
RBFNN-FFA	1745.3	1734.3	1723.8	1729.1	0.78	0.70	0.39	0.42
ANFI	1767.2	1763.2	1734.2	1731.2	0.77	0.69	0.40	0.45
MLP	1787.1	1765.2	1756.2	1744.2	0.76	0.67	0.41	0.47
RBFNN	1799.2	1789.1	1769.3	1767.2	0.75	0.65	0.43	0.49

$$\frac{S}{N} \text{ ratio} = -10 \log(\text{objective}(\text{function}))^2 \quad (33)$$

where the optimal value of random parameters has the highest S/N ratio. Figure 8 depicts the computed S/N ratio for different parameters in the four optimization algorithms tested here.

3.2 Talar station

For Talar station, the best results with the training dataset were obtained when ANFIS-SCA was used (RMSE: 934.2

ton/day, MAE: 912.3 ton/day, NSE: 0.93, PBIAS: 0.14) and the worst results were obtained when RBFNN was used (RMSE: 1789.10 ton/day, MAE: 1765.2 ton/day, NSE: 0.77, PBIAS: 0.36) (Table 3). The MLP-SCA was the best second model, and the hybrid and stand-alone MLP outperformed the hybrid and stand-alone RBFNN models (Table 3). Comparison of results obtained using the optimization algorithms revealed that SCA provided the best results and FFA the worst results.

The best results with the testing dataset for Talar station were also obtained with ANFIS-SCA (RMSE: 1423.2

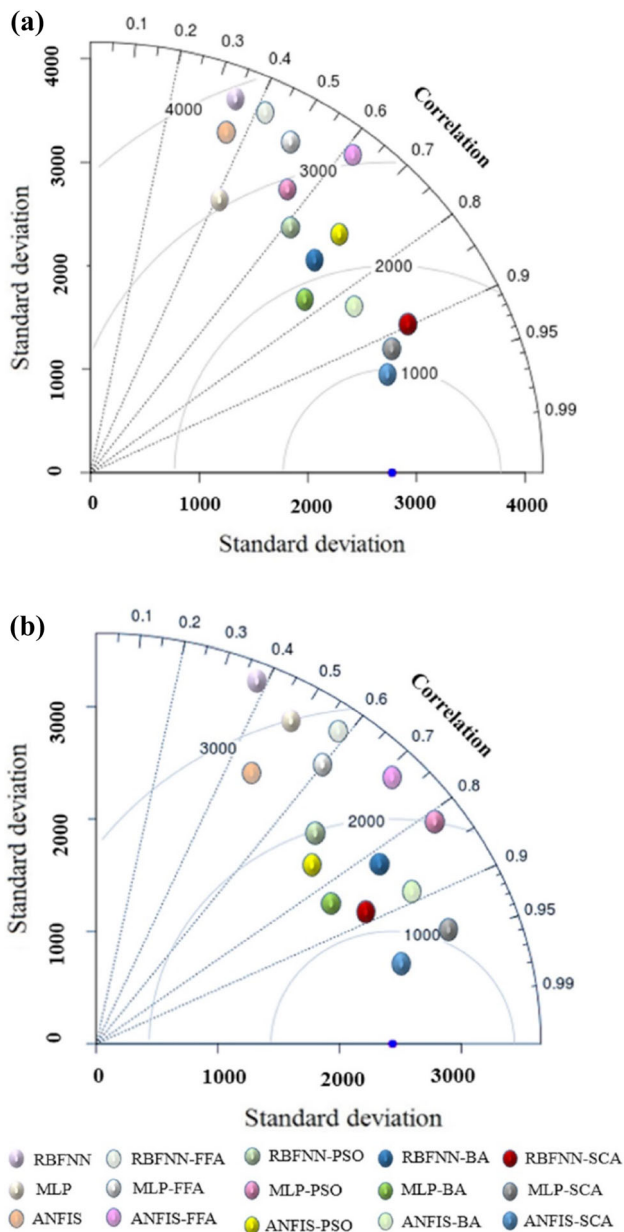


Fig. 9 Taylor diagram for the different hybrid and stand-alone models (using whole dataset) for **a** Talar station and **b** Kasilian station

ton/day, MAE: 1412.10 ton/day, NSE: 0.92, PBIAS: 0.16) (Table 3). Based on the RMSE, MAE, and PBIAS values, in the test stage the stand-alone ANFIS, MLP, and RBFNN models were more accurate than the hybrid ANFIS, MLP, and RBFNN models. Adding SCA decreased the RMSE of ANFIS, MLP, and RBFNN by 20%, 21%, and 22%, respectively. The testing results indicated that the hybrid and stand-alone MLP was better than the hybrid and stand-alone RBFNN models.

Table 4 Uncertainty results (*r*, *p*) for the different hybrid and stand-alone models

Model	<i>r</i> index		<i>p</i> index	
	Talar	Kasilian	Talar	Kasilian
ANFIS-SCA	0.14	0.12	0.94	0.95
MLP-SCA	0.17	0.14	0.90	0.94
RBFNN-SCA	0.19	0.16	0.89	0.93
ANFIS-BA	0.21	0.19	0.87	0.91
MLP-BA	0.25	0.20	0.86	0.90
RBFNN-BA	0.26	0.23	0.84	0.89
ANFIS-PSO	0.28	0.24	0.84	0.88
MLP-PSO	0.32	0.26	0.83	0.87
RBFNN-PSO	0.33	0.28	0.82	0.86
ANFIS-FFA	0.34	0.31	0.82	0.85
MLP-FFA	0.35	0.33	0.81	0.82
RBFNN-FFA	0.36	0.35	0.80	0.81
ANFIS	0.37	0.36	0.79	0.80
MLP	0.42	0.37	0.76	0.78
RBFNN	0.45	0.39	0.74	0.76

3.3 Kasilian station

For Kasilian station, in the training stage ANFIS-SCA was the best performing model (RMSE: 898.1 ton/day, MAE: 712.3 ton/day, NSE: 0.95, PBIAS: 0.12) and RBFNN was the worst (RMSE: 1655.6 ton/day, MAE: 1645.2 ton/day, NSE: 0.82, PBIAS: 0.39) (Table 3). The hybrid and stand-alone MLP models outperformed the stand-alone and hybrid RBFNN models during the training phase, while ANFIS-SCA outperformed MLP-SCA and RBFNN-SCA in terms of precision. Overall, the NSE, MAE, PBIAS, and NSE values for SCA proved its superiority among the optimization algorithms tested, while FFA gave the worst results. The performance of the hybrid ANFIS, MLP, and RBFNN models surpassed that of their stand-alone counterpart in the training stage.

In the testing phase, ANFIS-SCA again provided the best results (RMSE: 1412.10 ton/day, MAE: 1403.4 ton/day, NSE: 0.92; PBIAS: 0.14), and RBFNN again exhibited the worst results (RMSE: 1789.1 ton/day, MAE: 1767.2 ton/day, NSE: 0.65; PBIAS: 0.49) (Table 3). The hybrid ANFIS, MLP, and RBFNN models outperformed the stand-alone ANFIS, MLP, and RBFNN models. Based on the assessment statistics for Kasilian Station, it can be said that the SCA was the most accurate optimization algorithm and, as in the training phase, FFA gave the worst results) (Table 3). The evaluation criteria also confirmed the superiority of ANFIS-SCA, followed by the MLP-SCA, in comparison with RBFNN-SCA.

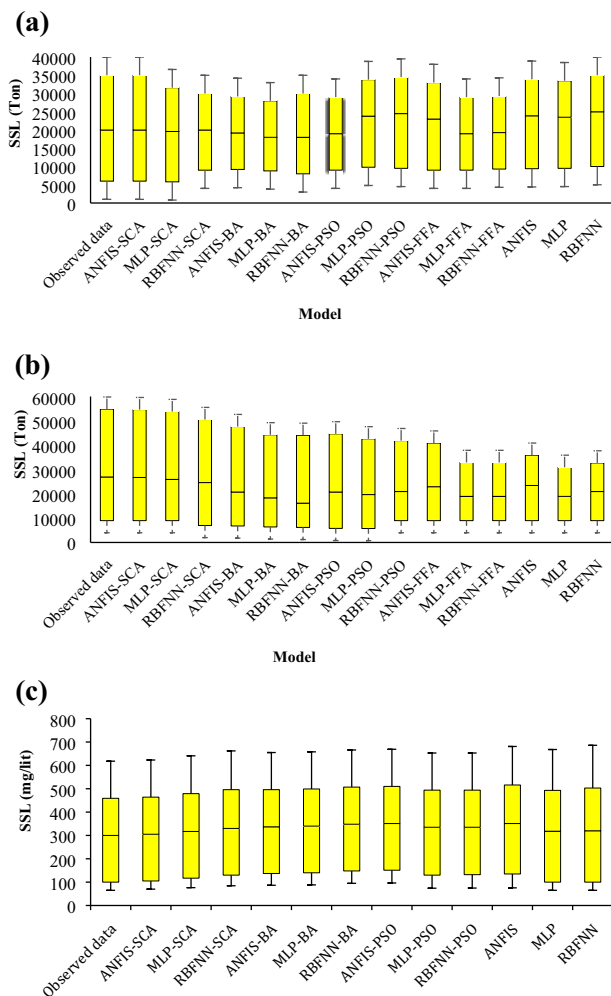


Fig. 10 Box plots of suspended sediment load (SSL) values obtained with the different hybrid and stand-alone models for **a** Talar and **b** Kasilian, **c** Egel Creek Basin stations

In order to visually summarize how closely the proposed models matching the observed values, Taylor diagrams were used to display the match between observed data and the output of the models in terms of their RMSE, standard deviation, and correlation. The Taylor diagram for Talar station, in which statistics for the 15 models (see Table 3) were calculated and a colored circle was assigned to each model, is shown in Fig. 8a. The position of each circle appearing in the diagram quantifies how that model's estimated SSL matched measured data, where the centered RMSE is proportional to the distance from the reference point on the horizontal axis as observed data. The whole dataset was used to plot the Taylor diagrams. The results revealed that for ANFIS-SCA, MLP-SCA, and RBFNN-SCA, the centered RMSE was 1000, 1050, and 1189 m, respectively. The hybrid soft computing models resulted in lower RMSE than the stand-alone models (Fig. 9a).

The Taylor diagram for Kasilian station is shown in Fig. 8b. It indicated that ANFIS-SCA and MLP-SCA predictions gave the best match with observed data and that the ANFIS-SCA model had a higher Taylor correlation and lower RMSE than the other models. The RBFNN model had the highest RMSE. When using the hybrid MLP, RBFNN, and ANFIS model, the Taylor correlation increased from 0.4 to 0.97. The highest RMSE was found for the stand-alone and hybrid MLP models (980–3000 ton/day) and the stand-alone and hybrid RBFNN models (1100 to 3300 ton/day) (Fig. 9b).

3.4 Uncertainty analysis of models and box plots

Comparison of models in terms of the selected indices (p and r) showed that ANFIS-SCA provided better r (0.12) and p (0.95) values for both Talar and Kasilian stations (Table 4). The hybrid ANFIS, MLP, and RBFNN models gave better p and r values than the stand-alone ANFIS, MLP, and RBFNN models. In addition, RBFNN-FFA had the lowest p (0.76) and highest r (0.39) among the hybrid models. For Talar station, the hybrid and stand-alone ANFIS models gave better r and p values than the hybrid and stand-alone MLP and RBFNN models (Table 4). Figure 10a, b shows the box plots for different models at Talar and Kasilian stations. The results indicated that ANFIS-SCA, MLP-SCA, and RBFNN-SCA most closely matched the observed SSL, outperforming the stand-alone ANFIS, MLP, and RBFNN models at both stations.

Overall, this study showed that the hybrid ANFIS-SCA model has good ability for predicting SSL in rivers. However, different climate parameters affected the SSL values obtained (Table 2), so follow-up studies should predict SSL for future periods using climate models and scenarios describing projected changes in meteorological parameters such as temperature and rainfall.

3.4.1 The analysis for the Egel Creek river basin

Table 2 indicates that the first three components (PC_1 , PC_3 , and PC_3) have greater values of participation. Furthermore, $SSL(t-1)$, $R(t-1)$, and $Q(t-1)$ data are more significant for all three components compared to other data. Thus, the first three components were selected as input to the models. Table 5 shows that the ANFIS-SCA model reduced RMSE by 15% and 21% compared to the MLP-SCA and RBFNN-SCA models in the training phase. Comparing models performance indicated that the ANFIS-SCA model could decrease MAE error compared to ANFIS-BA, ANFIS-PSO, ANFIS-FFA, and ANFIS models by 18%, 32%, 37%, and 49% in the training phase, respectively. Comparing the performance of the models showed that the ANFIS-SCA model with the highest value

Table 5 Performance of the different hybrid and stand-alone models for the training and testing phases for the *Eagel Creek Basin*

Model	Train						
	RMSE (mg/L)		MAE (mg/L)		NSE		PBIAS
	RMSE	MAE	NSE	PBIAS	<i>r</i>	<i>p</i>	R ²
ANFIS-SCA	12.23	11.10	0.98	0.14	0.12	0.94	0.95
MLP-SCA	14.35	12.23	0.97	0.16	0.14	0.91	0.92
RBFNN-SCA	15.36	14.23	0.95	0.18	0.16	0.90	0.90
ANFIS-BA	14.78	14.35	0.94	0.20	0.18	0.87	0.85
MLP-BA	15.67	14.87	0.92	0.22	0.20	0.85	0.84
RBFNN-BA	16.67	15.12	0.91	0.24	0.22	0.82	0.80
ANIFS-PSO	18.12	16.12	0.90	0.26	0.25	0.79	0.78
MLP-PSO	18.25	17.23	0.89	0.28	0.27	0.77	0.76
RBFNN-PSO	19.12	18.24	0.88	0.32	0.29	0.75	0.72
ANFIS-FFA	19.14	18.45	0.86	0.33	0.30	0.72	0.68
MLP-FFA	20.21	19.12	0.85	0.34	0.31	0.68	0.66
RBFNN-FFA	22.23	20.12	0.82	0.36	0.32	0.67	0.65
ANFIS	23.45	21.23	0.80	0.39	0.33	0.66	0.64
MLP	24.56	22.10	0.76	0.40	0.34	0.62	0.62
RBFNN	25.67	22.12	0.77	0.42	0.35	0.60	0.58

Model	Test						
	RMSE	MAE	NSE	PBIAS	<i>r</i>	<i>p</i>	R ²
ANFIS-SCA	16.12	15.78	0.96	0.16	0.14	0.90	0.94
MLP-SCA	17.23	16.12	0.94	0.18	0.18	0.89	0.90
RBFNN-SCA	17.45	16.98	0.90	0.20	0.20	0.87	0.89
ANFIS-BA	17.68	17.02	0.89	0.22	0.22	0.85	0.85
MLP-BA	19.23	18.23	0.87	0.25	0.24	0.84	0.84
RBFNN-BA	20.12	19.12	0.85	0.29	0.26	0.82	0.80
ANIFS-PSO	21.12	20.12	0.82	0.31	0.30	0.80	0.76
MLP-PSO	22.34	21.12	0.80	0.32	0.32	0.76	0.75
RBFNN-PSO	24.45	22.34	0.78	0.35	0.33	0.65	0.70
ANFIS-FFA	25.67	24.67	0.76	0.39	0.35	0.60	0.65
MLP-FFA	27.62	27.65	0.74	0.42	0.37	0.59	0.60
RBFNN-FFA	28.63	27.69	0.70	0.45	0.39	0.58	0.59
ANFI	29.12	28.14	0.68	0.49	0.40	0.57	0.58
MLP	30.12	29.12	0.65	0.51	0.42	0.56	0.57
RBFNN	31.12	30.12	0.62	0.55	0.44	0.55	0.56

of *p* and the lowest value of *r* had less uncertainty than other models in the training phase. Moreover, the findings indicated that RBFNN model with 0.77 NSE had the weakest performance in the training phase. Examining the performance of the models in the training phase according to PBIAS value indicated that the MLP-SCA model had less PBIAS than other MLP models, showing better performance of the MLP-SCA model than other models. Comparing the performance of the models in the test phase indicated that the ANFIS-SCA model reduced RMSE value by 8.8%, 24%, 38%, and 45%, respectively, compared to

the ANFIS-BA, ANFIS-PSO, ANFIS-FFA, and ANFIS in the test phase. Additionally, the results indicated that the RBFNN model with NSE and PBIAS of 0.62 and 0.55 has the weakest performance among the models in the test phase. One can see that the hybrid ANFIS models had higher and lower *p* and *r* values than the ANFIS model, showing less uncertainty of hybrid ANFIS models compared to the ANFIS model. Ultimately, Fig. 10c indicates that ANFIS-SCA box diagram was more in line with the observational data compared to other models. Figure 10c indicates that ANFIS, MLP, and RBFNN models are less

accurate than models using optimization algorithms. Hence, the performance of the models for the second case study indicated that ANFIS-SCA model had better accuracy in the present study. In Table 5, the performance of the different hybrid and stand-alone models for the training and testing phases is presented for the Egel Creek Basin.

4 Conclusions

The knowledge of suspended sediment load modeling in rivers is excessive, as it results from soil erosion and plays a key role in watershed management, river morphology, and the operation of hydraulic structures. The current research studied any possibility of evolutionary soft computing approaches in suspended sediment load modeling. However, soft computing approaches such as the ANFIS, MLP, and RBFNN models are widely used to estimate SSL, but their output is not sufficiently accurate for basin management. In this study, four optimization algorithms (SCA, PSO, BA, and FFA) were used to train the ANFIS, MLP, and RBFNN models for suspended sediment load prediction at the basin scale (Talar and Egel Creek Basins located in northern Iran and central part of USA). The second case study demonstrated that the ANFIS-SCA model could decrease MAE error compared to ANFIS-BA, ANFIS-PSO, ANFIS-FFA, and ANFIS models by 18%, 32%, 37%, and 49% in the training phase, respectively. However, different climate parameters affected the SSL value, so future studies should predict SSL for future periods using models and scenarios describing future changes in climate. Each optimization algorithm in the study with high accuracy and appropriate convergence speed showed a very high capacity for solving optimization problems. The conclusions are as follows:

- Novel optimized models had an important scientific contribution to the development of a powerful model for suspended sediment load prediction at the watershed scale. The sine–cosine algorithm (SCA) optimizer gave strong predictive capacities to the (multilayer perceptron (MLP), adaptive neuro-fuzzy system (ANFIS), and radial basis function neural network (RBFNN)).
- Among the optimized models, ANFIS-SCA showed the best performance in the training and testing phases for both stations, while RBFNN showed the lowest accuracy.
- Optimization of the models using SCA has decreased the RMSE by 20%, 21%, and 22% for ANFIS, MLP, and RBFNN, respectively.
- The uncertainty outputs (based on the uncertainty in sequential uncertainty fitting (SUFI-2)) indicated that the hybrid ANFIS, MLP, and RBFNN models were the

most accurate (lowest r index, highest p index) of the models tested. Overall, ANFIS-SCA showed a good ability for predicting SSL.

- This study can help as a basic research for future studies and other regions (other optimization algorithms or soft computing models) seeking suspended sediment load prediction in a watershed scale using optimized models.

Funding Open access funding provided by University of Oulu including Oulu University Hospital.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Research involving human participants and/or animals This article does not contain any studies with human participants and/or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Adib A, Mahmoodi A (2017) Prediction of suspended sediment load using ANN GA conjunction model with Markov chain approach at flood conditions. *KSCE J Civ Eng* 21(1):447–457
- Afan HA, El-shafie A, Mohtar WHMW, Yaseen ZM (2016) Past, present and prospect of an Artificial Intelligence (AI) based model for sediment transport prediction. *J Hydrol* 541:902–913
- Ahmad ST, Kumar KP (2016) Radial basis function neural network nonlinear equalizer for 16-QAM coherent optical OFDM. *IEEE Photonics Technol Lett* 28(22):2507–2510
- Akrami SA, El-Shafie A, Jaafar O (2013) Improving Rainfall Forecasting Efficiency Using Modified Adaptive Neuro-Fuzzy Inference System (MANFIS). *Water Resour Manag* 27(9):3507–3523
- Azamathulla HM, Ghani AA, Fei SY (2012) ANFIS-based approach for predicting sediment transport in clean sewer. *Appl Soft Comput* 12(3):1227–1230
- Bezák N, Mikoš M, Šraj M (2014) Trivariate frequency analyses of peak discharge, hydrograph volume and suspended sediment concentration data using copulas. *Water Resour Manag* 28(8):2195–2212
- Bhowmik M, Muthukumar P, Anandalakshmi R (2019) Experimental based multilayer perceptron approach for prediction of

- evacuated solar collector performance in humid subtropical regions. *Renew Energy* 143:1566–1580
- Chang FJ, Lai HC (2014) Adaptive neuro-fuzzy inference system for the prediction of monthly shoreline changes in northeastern Taiwan. *Ocean Eng* 84:145–156
- Choubin B, Darabi H, Rahmati O, Sajedi-Hosseini F, Kløve B (2018) River suspended sediment modelling using the CART model: a comparative study of machine learning techniques. *Sci Total Environ* 615:272–281
- Cigizoglu HK (2004) Estimation and forecasting of daily suspended sediment data by multi-layer perceptrons. *Adv Water Resour* 27(2):185–195
- Cobaner M, Unal B, Kisi O (2009) Suspended sediment concentration estimation by an adaptive neuro-fuzzy and neural network approaches using hydro-meteorological data. *J Hydrol* 367(1–2):52–61
- Darabi H, Shahedi K, Solaimani K, Miryaghoubzadeh M (2014) Prioritization of subwatersheds based on flooding conditions using hydrological model, multivariate analysis and remote sensing technique. *Water Environ J* 28(3):382–392
- Downs PW, Cui Y, Wooster JK, Dusterhoff SR, Booth DB, Dietrich WE, Sklar LS (2009) Managing reservoir sediment release in dam removal projects: An approach informed by physical and numerical modelling of non-cohesive sediment. *Int J River Basin Manag* 7(4):433–452
- Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: *MHS'95. Proceedings of the sixth international symposium on micro machine and human science*, pp 39–43
- Ehteram M, Karami H, Mousavi SF, El-Shafie A, Amini Z (2017) Optimizing dam and reservoirs operation based model utilizing shark algorithm approach. *Knowl Based Syst* 122(26):38
- Ehteram M, Ahmed AN, Latif SD, Huang YF, Alizamir M, Kisi O, El-Shafie A (2020) Design of a hybrid ANN multi-objective whale algorithm for suspended sediment load prediction. *Environ Sci Pollut Res* 28:1–16
- Emamgholizadeh S, Demneh RK (2019) A comparison of artificial intelligence models for the estimation of daily suspended sediment load: a case study on the Telar and Kasilian rivers in Iran. *Water Supply* 19(1):165–178
- Erol R, Oğulata SN, Şahin C, Alparslan ZN (2008) A radial basis function neural network (RBFNN) approach for structural classification of thyroid diseases. *J Med Syst* 32(3):215–220
- Fiyadh SS, AlSaadi MA, Jaafar WZ, AlOmar MK, Fayaed SS, Mohd NS, Hin LS, El-Shafie A (2019) Review on heavy metal adsorption processes by carbon nanotubes. *J Clean Prod* 230:783–793
- Gholami A, Bonakdari H, Zaji AH, Michelson DG, Akhtari AA (2016) Improving the performance of multi-layer perceptron and radial basis function models with a decision tree model to predict flow variables in a sharp 90 bend. *Appl Soft Comput* 48:563–583
- Guo C, Jin Z, Guo L, Lu J, Ren S, Zhou Y (2020) On the cumulative dam impact in the upper Changjiang River: Streamflow and sediment load changes. *CATENA* 184:104250
- Haghighi AT, Darabi H, Shahedi K, Solaimani K, Kløve B (2019) A scenario-based approach for assessing the hydrological impacts of land use and climate change in the Marboreh Watershed. *Iran. Environ Model Assess* 25:1–17
- Halecki W, Kruk E, Ryczek M (2018) Estimations of nitrate nitrogen, total phosphorus flux and suspended sediment concentration (SSC) as indicators of surface-erosion processes using an ANN (Artificial Neural Network) based on geomorphological parameters in mountainous catchments. *Ecol Indic* 91:461–469
- Han HG, Qiao JF, Chen QL (2012) Model predictive control of dissolved oxygen concentration based on a self-organizing RBF neural network. *Control Eng Pract* 20(4):465–476
- Himanshu SK, Pandey A, Yadav B (2017) Assessing the applicability of TMPA-3B42V7 precipitation dataset in wavelet-support vector machine approach for suspended sediment load prediction. *J Hydrol* 550:103–117
- Ibrahim S, Choong CE, El-Shafie A (2019) Sensitivity analysis of artificial neural networks for just-suspension speed prediction in solid-liquid mixing systems: performance comparison of MLPNN and RBFNN. *Adv Eng Inform* 39:278–291
- Kaveh K, Kaveh H, Bui MD, Rutschmann P (2020) Long short-term memory for predicting daily suspended sediment concentration. *Eng Comput*. <https://doi.org/10.1007/s00366-019-00921-y>
- Kavian A, Mohammadi M, Gholami L, Rodrigo-Comino J (2018) Assessment of the spatiotemporal effects of land use changes on runoff and nitrate loads in the Talar River. *Water* 10(4):445
- Kim TW, Valdés JB (2003) Nonlinear model for drought forecasting based on a conjunction of wavelet transforms and neural networks. *J Hydrol Eng* 8(6):319–328
- Kisi O, Dailr AH, Cimen M, Shiri J (2012) Suspended sediment modeling using genetic programming and soft computing techniques. *J Hydrol* 450:48–58
- Kong C, Wang H, Li D, Zhang Y, Pan J, Zhu B, Luo Y (2016) Quality changes and predictive models of radial basis function neural networks for brined common carp (*Cyprinus carpio*) fillets during frozen storage. *Food Chem* 201:327–333
- Kumar D, Pandey A, Sharma N, Flügel WA (2016) Daily suspended sediment simulation using machine learning approach. *CATENA* 138:77–90
- Kumar N, Singh SK, Srivastava PK, Narsimlu B (2017) SWAT Model calibration and uncertainty analysis for streamflow prediction of the Tons River Basin, India, using Sequential Uncertainty Fitting (SUFI-2) algorithm. *Model Earth Syst Environ* 3(1):30
- Lafdani EK, Nia AM, Ahmadi A (2013) Daily suspended sediment load prediction using artificial neural networks and support vector machines. *J Hydrol* 478:50–62
- Liu QJ, Zhang HY, Gao KT, Xu B, Wu JZ, Fang NF (2019) 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* 179:107–118
- Lu H, Meng Y, Yan K, Gao Z (2019) Kernel principal component analysis combining rotation forest method for linearly inseparable data. *Cogn Syst Res* 53:111–122
- Malmon DV, Dunne T, Reneau SL (2002) Predicting the fate of sediment and pollutants in river floodplains. *Environ Sci Technol* 36(9):2026–2032
- McCarney-Castle K, Childress TM, Heaton CR (2017) Sediment source identification and load prediction in a mixed-use Piedmont watershed, South Carolina. *J Environ Manag* 185:60–69
- Melesse AM, Ahmad S, McClain ME, Wang X, Lim YH (2011) Suspended sediment load prediction of river systems: An artificial neural network approach. *Agric Water Manag* 98(5):855–866
- Merkhali SP, Ehteshami M, Sadrnejad SA (2015) Assessment quality of a nonuniform suspended sediment transport model under unsteady flow condition (case study: Aras River). *Water Environ J* 29(4):489–498
- Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. *Knowl-Based Syst* 96:120–133
- Mozdgrir A, Mahdavi I, Badeleh IS, Solimanpur M (2013) Using the Taguchi method to optimize the differential evolution algorithm parameters for minimizing the workload smoothness index in simple assembly line balancing. *Math Comput Model* 57(1–2):137–151

- Nourani V, Andalib G (2015) Daily and monthly suspended sediment load predictions using wavelet based artificial intelligence approaches. *J Mountain Sci* 12(1):85–100
- Park I, Choi J, Lee MJ, Lee S (2012) Application of an adaptive neuro-fuzzy inference system to ground subsidence hazard mapping. *Comput Geosci* 48:228–238
- Pektaş AO, Doğan E (2015) Prediction of bed load via suspended sediment load using soft computing methods. *Geofizika* 32(1):27–46
- Pirnia A, Darabi H, Choubin B, Omidvar E, Onyutha C, Haghghi AT (2019a) Contribution of climatic variability and human activities to stream flow changes in the Haraz River basin, northern Iran. *J Hydro-environ Res* 25:12–24
- Pirnia A, Golshan M, Darabi H, Adamowski J, Rozbeh S (2019b) Using the Mann-Kendall test and double mass curve method to explore stream flow changes in response to climate and human activities. *J Water Clim Change* 10(4):725–742
- Pizzuto J (2020) Suspended sediment and contaminant routing with alluvial storage: new theory and applications. *Geomorphology* 352:106983
- Rajae T, Mirbagheri SA, Zounemat-Kermani M, Nourani V (2009) Daily suspended sediment concentration simulation using ANN and neuro-fuzzy models. *Sci Total Environ* 407(17):4916–4927
- Ren J, Zhao M, Zhang W, Xu Q, Yuan J, Dong B (2020) Impact of the construction of cascade reservoirs on suspended sediment peak transport variation during flood events in the Three Gorges Reservoir. *CATENA* 188:104409
- Romano G, Abdelwahab OM, Gentile F (2018) Modeling land use changes and their impact on sediment load in a Mediterranean watershed. *CATENA* 163:342–353
- Sabour MR, Movahed SMA (2017) Application of radial basis function neural network to predict soil sorption partition coefficient using topological descriptors. *Chemosphere* 168:877–884
- Sahin S (2012) An aridity index defined by precipitation and specific humidity. *J Hydrol* 444:199–208
- Salih SQ, Sharafati A, Khosravi K, Faris H, Kisi O, Tao H et al (2020) River suspended sediment load prediction based on river discharge information: application of newly developed data mining models. *Hydrol Sci J* 65(4):624–637
- Samanta S, Suresh S, Senthilnath J, Sundararajan N (2019) A new neuro-fuzzy inference system with dynamic neurons (nfis-dn) for system identification and time series forecasting. *Appl Soft Comput* 82:105567
- Samantaray S, Ghose DK (2020) Assessment of suspended sediment load with neural networks in arid watershed. *J Inst Eng India Ser A* 101:371–380
- Shamaei E, Kaedi M (2016) Suspended sediment concentration estimation by stacking the genetic programming and neuro-fuzzy predictions. *Appl Soft Comput* 45:187–196
- Shojaeazadeh SA, Nikoo MR, McNamara JP, AghaKouchak A, Sadegh M (2018) Stochastic modeling of suspended sediment load in alluvial rivers. *Adv Water Resour* 119:188–196
- Singh A, Imtiyaz M, Isaac RK, Denis DM (2014) Assessing the performance and uncertainty analysis of the SWAT and RBNN models for simulation of sediment yield in the Nagwa watershed. *India Hydrolog Sci J* 59(2):351–364
- Talebi A, Mahjoobi J, Dastorani MT, Moosavi V (2017) Estimation of suspended sediment load using regression trees and model trees approaches (Case study: Hyderabad drainage basin in Iran). *ISH J Hydrol Eng* 23(2):212–219
- Vafakhah M (2013) Comparison of cokriging and adaptive neuro-fuzzy inference system models for suspended sediment load forecasting. *Arab J Geosci* 6(8):3003–3018
- Van Dao D, Jaafari A, Bayat M, Mafi-Gholami D, Qi C, Moayedi H, Luu C (2020) A spatially explicit deep learning neural network model for the prediction of landslide susceptibility. *CATENA* 188:104451
- Wang X, Shi Z, Shi Y, Ni S, Wang R, Xu W, Xu J (2018) Distribution of potentially toxic elements in sediment of the Anning River near the REE and V-Ti magnetite mines in the Panxi Rift, SW China. *J Geochem Explor* 184:110–118
- Yang CT, Marsooli R, Aalami MT (2009) Evaluation of total load sediment transport formulas using ANN. *Int J Sedim Res* 24(3):274–286
- Yurdusev MA, Firat M (2009) Adaptive neuro fuzzy inference system approach for municipal water consumption modeling: An application to Izmir. Turkey *J Hydrol* 365(3–4):225–234
- Zhang X, Fichot CG, Baracco C, Guo R, Neugebauer S, Bengtsson Z, Fagherazzi S (2020) Determining the drivers of suspended sediment dynamics in tidal marsh-influenced estuaries using high-resolution ocean color remote sensing. *Remote Sens Environ* 240:111682
- Zhao G, Pang B, Xu Z, Xu L (2020) A hybrid machine learning framework for real-time water level prediction in high sediment load reaches. *J Hydrol* 581:124422

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Hamid Darabi¹ · Sedigheh Mohamadi² · Zahra Karimidastenaee¹ · Ozgur Kisi³ · Mohammad Ehteram⁴ · Ahmed ELShafie⁵ · Ali Torabi Haghghi¹

✉ Hamid Darabi
hamid.darabi@oulu.fi

¹ Water, Energy and Environmental Engineering Research Unit, University of Oulu, P.O. Box 4300, 90014 Oulu, Finland

² Department of Ecology, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran

³ Department of Civil Engineering, School of Technology, Ilia State University, 0162 Tbilisi, Georgia

⁴ Department of Water Engineering and Hydraulic Structures, Faculty of Civil Engineering, Semnan University, Semnan, Iran

⁵ Department of Civil Engineering, Faculty of Engineering, University of Malaya, Kuala Lumpur, Malaysia