Flood Routing by the Muskingum Method and Neural Network

Marjan Moazamnia

Dept. of Civil Engineering, Faculty of Engineering, University of Bonab, Bonab, East Azerbaijan, Iran

Received date**: 03.05.2025**; Accepted date**: 29.05.2025**; Published date: **28.06.2025**

Turkish Journal of Hydraulic, Vol: **9**, Number : **1**, Page, **25-32**, **(2025)**

e-ISSN: **2636-8382**

SLOI: http://www.dergipark.org.tr

Corresponding author e-mail: m.moazamnia@ubonab.ac.ir

Abstract

Floods have consistently been one of the most significant natural disasters affecting humans. In a country like Iran, their impact is particularly pronounced due to the irregular patterns of rainfall both in space and time. Flood routing is a crucial aspect of hydraulic engineering, as it enables the prediction of how floods will rise and recede at specific points along a river. Various techniques and methods are employed to address routing problems. This Manuscript explores routing using Muskingum's method, the least squares error method, and neural networks. First, three proposed neural network models with different transfer functions were evaluated to identify the best-performing model. The results were then compared using the least squares method and validated against the model proposed by Choudhury and Sankarasubramanian (2009). Ultimately, both models yielded acceptable results; however, considering the RMSE values, the least squares error method's results are closer to those proposed by Choudhury and Sankarasubramanian (2009).

Keywords: *Flood routing, Muskingum, Least Squares Error, Neural Network, Transfer Function*

1. INTRODUCTION

Flood flow in a channel refers to the movement of a wave, making it crucial for engineers to accurately predict how flood levels will rise and fall at specific locations along the channel. Flood routing involves a series of operations that use the known upstream flow hydrograph to determine the downstream flow hydrograph. In other words, computational operations for analyzing flood patterns in streams (channels and rivers) involve predicting changes in hydraulic variables, flow geometry, and flood waveforms overtime at one or more points along the stream (Mirzazade, 2013). Using routing techniques and a single-point flood hydrograph, the desired flood height can be determined at any location along the river's course. Various hydraulic and hydrological methods are available to route floods. If data and statistics of output sections are not needed, hydraulic methods can be used, where output data is obtained through hydraulic routing at any section. Flood routing by hydrological methods is relatively simple and reasonably accurate but requires multiple inflow and outflow hydrograph data.

The most common river routing method is the Muskingum method. This method was first developed by McCarthy for flood control studies of the Muskingum River basin in Ohio in 1938. This method uses the continuity equation to perform river routing. Perumal (1994) derived the Muskingum method with variable parameters for flood wave routing in fixed-section channels directly from the Saint-Venant equations by assuming a constant water level gradient along a small span of the channel and establishing a steady flow between the depth in the middle of the span and the discharge at a section downstream. Mohan (1997) proposed a genetic algorithm for estimating the parameters of two nonlinear Muskingum methods.

This study compared the results of the proposed genetic algorithm with those obtained from nonlinear least squares and conjugate gradient regression methods. Unlike other techniques, the genetic algorithm does not require an initial estimation of parameters. The application of this method to the nonlinear relationship between storage and flow demonstrated that the genetic algorithm effectively estimates the parameters of the nonlinear model.

Al-Humoud and Esen (2006) used an approximate and straightforward method to determine the coefficients of the Muskingum method. This method is estimated based on calculating the slope of the inflow and outflow hydrographs at their junction. Chu and Chang (2009) used an adaptive inference system in the MATLAB environment to determine the flood trend using the Muskingum method. They compared the results obtained from the neural-fuzzy network with the genetic algorithm. They concluded that the values obtained from the fuzzy neural network can be used in the Muskingum method and have a better match with the observational data than other methods. Easa (2013) used the variable parameter method in flood routing. In his study, he used three flood hydrographs and considered the exponential parameter in the nonlinear Muskingum method as a variable. In his research, the exponential parameter changes with the amount of inflow. The inflow is divided into five parts, and a separate exponential parameter is considered for each part. Niazkar and Afzali (2014) used the modified Honey Bee Mating Optimization algorithm (HBMO) to find the parameters of the Muskingum method.

The primary advantage of this approach is its ability to quickly reach an optimal value across a wide range of parameters. Moghaddam et al. (2016) initially estimated the parameters of the nonlinear Muskingum method using Particle Swarm Optimization (PSO). They then applied this method to a new form of the four-parameter Muskingum method, utilizing three sample hydrographs and one real hydrograph from Iran. Their findings demonstrated that, although the new Muskingum method was more complex, it provided a better fit to the observational data, particularly for hydrographs with multiple flow peaks. katipoğlu and Sarıgöl (2023) for flood routing prediction applying empirical model decomposition (EMD) and neural networks. This study showed that the EMD model can improve the performance of machine learning models, and the EMD model was the most successful algorithm in flood routing computation. The aim of the study of Sari Sarıgöl (2024) is to compare the performance of machine learning, deep learning, and hybrid algorithms for flood routing prediction models in the Büyük Menderes River. In the research deep learning model Long-Short Term Memory (LSTM), machine learning model Artificial Neural Network (ANN), and hybrid machine learning models empirical mode decomposition (EMD)-ANN, and particle swarm optimization (PSO)-ANN algorithms were compared to forecast the flood routing results in two discharge observation stations in the Büyük Menderes river. The results showed that the hybrid algorithm PSO-ANN was the most successful in forecasting flood routing results among other models. In the present study, flood routing was performed using the Muskingum method by the least squares method and neural networks. There have been relatively limited studies in the field of reservoir routing using artificial intelligence models, and this is one of the innovations of the present article.

2. METHODOLOGY

2.1. Muskingum Method

Flood routing methods can be divided into hydraulic and hydrological. In hydraulic routing, the continuity equation and the equation of motion of unsteady flow are used, and in hydrological routing, which is a standard method in water engineering, the equation of motion is generally ignored, and the continuity equation for a control volume is used as follows:

$$I - O = \frac{ds}{dt} \tag{1}$$

Where I, O, and S are the inflow and outflow rates and the storage volume of the control volume, respectively. The governing relationship for the Muskingum method, which is obtained through the scaling curve equation, is in the form of Equation 2.

$$S = KO - KX(I - O) \tag{2}$$

In the equation above, K represents the reservoir coefficient, while X denotes the weight coefficient. The parameter K reflects the river's travel time, which is determined by the river's length and the speed of the flood wave. In contrast, X signifies the influence of flood inflows and outflows on the river's storage volume, typically varying between 0 and 0.5. By assessing these two parameters, it is possible to calculate the flood outflow hydrograph for each flood event in the river. The following equation (Equation 3) are utilized for this calculation:

$$O_{t+1} = C_1 I_t + C_2 I_{t+1} + C_3 O_t \tag{3}$$

Where I_t and I_{t+1} are the flood inflow discharges at times t and t+1 from the flood inflow hydrograph, respectively, O_t and O_{t+1} the flood outflow discharges at times t and t+1 (from the flood outflow hydrograph), C_1, C_2 and C_3 are the routing coefficients. These coefficients are determined from equations (4-6):

$$C_1 = \frac{Kx + 0.5\Delta t}{K(1-x) + 0.5\Delta t} \tag{4}$$

$$C_2 = \frac{0.5\Delta t - Kx}{K(1-x) + 0.5\Delta t} \tag{5}$$

$$C_3 = \frac{K - Kx - 0.5\Delta t}{K(1-x) + 0.5\Delta t} \tag{6}$$

Which Δt is the time step of the calculations. Equation (3) is used iteratively to calculate the outflow flood hydrograph by determining the trending coefficients.

2.2. Flood estimation model by Choudhury and Sankarasubramanian

The equation proposed by Choudhury and Sankarasubramanian (2009) is represented by equation (7):

$$C_1 \alpha I_t + C_3 \beta O_t = -(1 - C_1 - C_3) I_{t+\Delta t} \tag{7}$$

In this equation, c_1 and c_3 are Muskingum parameters and α and β are the upstream hydrograph evolution parameters, which are functions of the upstream watershed, river branch, and sudden change characteristics such as storms. The parameters c_1, c_3, α and β for a branch of the river are obtained by minimizing the objective functions (equations (8-11)).

$$f(1) = \min \sum_{t=1}^{N-1} \left(i_{t+1} - \overline{i_{t+1}}\right)^2 \tag{8}$$

$$f(2) = \min \sum_{t=1}^{N-1} \left(q_{t+1} - \overline{q_{t+1}}\right)^2 \tag{9}$$

$$f(3) = \min \sum_{t=1}^{N-2} \left(i_{t+2} - \overline{i_{t+2}}\right)^2 \tag{10}$$

$$f(4) = \min \sum_{t=1}^{N-2} \left(q_{t+2} - \overline{q_{t+2}}\right)^2 \tag{11}$$

In the equations mentioned above, i and q represents the predicted flow in the upstream section, while $\bar{i}$ and $\bar{q}$ denotes the predicted flow in the downstream section. These correspond to the observed inflow and outflow values, respectively. The Muskingum parameters have been derived by minimizing these equations. When used for prediction in two future time steps, the obtained parameters will have prediction errors due to measurement and estimation errors. The error at time t+2 may be present in the error terms at time t+1. To estimate the possible error at the next time step during the flow predictions, two other objective functions are formed as given in equations 12 and 13.

$$f(5) = \min \sum_{t=1}^{N-2} \left(E_{t+2}^{i} - \theta^{i} E_{t+1}^{i}\right)^2 \tag{12}$$

$$f(6) = \min \sum_{t=1}^{N-2} \left(E_{t+2}^{q} - \theta^{q} E_{t+1}^{q}\right)^2 \quad (13)$$

In the above equation, E is the error indicator and θ is the parameter that defines the propagation of the error. The optimal estimation of parameters θ^i and θ^q defines a linear relation between the error at times t+1 and t+2 for the upstream and downstream stations, respectively. Minimizing the six objective functions mentioned (Equations 8 to 13) makes it possible to estimate the model parameters and the error. With the parameter values obtained for the river branch, the model may be used for long-range prediction of the upstream and downstream stations.

2.3. Least Squares Error Method

This method minimizes the difference between the observed and estimated storage. Therefore, we formulate and minimize the error function in the following form.

$$E = \sum_j \left[AI_j + BO_j + S_1 - S_j\right] \quad (14)$$

Where A=KX, B=K(1-X), S_1 is the initial storage, and S_j is the observed relative storage at the jth time step. N is the number of observed values of inflow, outflow, and relative storage for the period under study in the river (Choudhury and Sankarasubramanian (2009)). After taking the derivative of the error equation and setting it equal to zero, the error function reaches its minimum value. The values of A and B will be obtained by solving the matrix form of equation 15:

$$\begin{bmatrix} \sum I_j^2 & \sum I_j O_j \\ \sum I_j O_j & \sum O_j^2 \end{bmatrix} \times \begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} \sum I_j S_j \\ \sum O_j S_j \end{bmatrix} \quad (15)$$

The least squares method calculations were performed using a program written in MATLAB. The values of X and K were obtained as 0.237 and 1.631, respectively. Finally, based on these calculated values, the output hydrograph was calculated, the results of which are presented in the following graphs.

2.4. Artificial neural network

The concept of artificial neural networks was first introduced by Frank Rosenblatt in 1962 and then in 1986 by Rommelhart and McClelland with the invention and presentation of the perceptron model, which is modeled after the neurons in the human brain and simulates the intracellular behavior of brain neurons through mathematical functions defined. The computational weights in the communication lines of artificial neurons play the role of synapses in natural neurons. The breadth and flexibility of neural networks have made them widely used in problems of a predictive nature.

The working process of a simple single-layer network is shown in Figure 1. The following simple network consists of one input and one neuron, and the role of the transfer function (denoted by f) is observed. The input p applied to the neuron is weighted by multiplying it by the weight w, and the result is used as an input to the transfer function f, and the final output is obtained. The bias input is a constant value of 1, a tunable parameter of the neurons, not an input, and its use in the MATLAB software toolbox is optional. The bias value is added to the product of w.p. The main idea of neural networks is that by changing the values of w and b, the network adopts a correct behavior or decision.

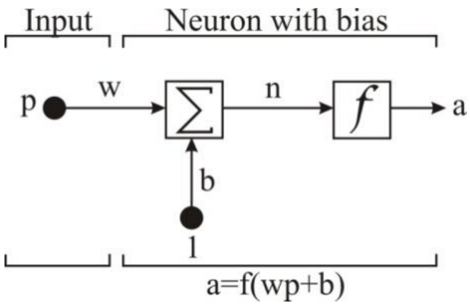

Figure 1. Structure of a single-layer neural network

A perceptron network has an input layer to apply inputs, a hidden layer, and an output layer to provide problem outputs. This type of network is usually trained using the backpropagation method. An example of this type of network is shown in Figure 2.

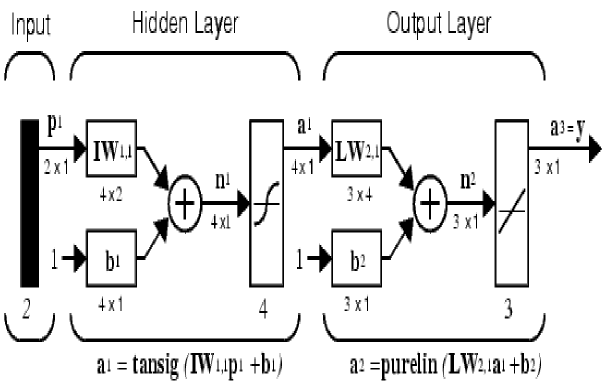

Figure 2. Structure of a multilayer perceptron with hidden neurons and output neurons with a linear function

The role of transfer functions in neural networks is to calculate the output of the layers from the input network. This review evaluated the performance of three different functions: logsig, tansig, and purelin. It is important to note that in multilayer networks, using the aforementioned transfer functions has been very common, but other transfer functions can also be used if desired. The diagram of these functions is given in Figure 3.

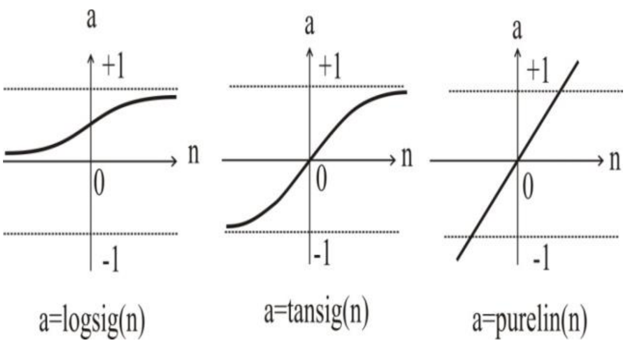

Figure 3. Diagram of transfer functions used in the neural network (Agami et al., 2009)

Given that the range of the tansig function is in the range of 1 and -1 and the range of the logsig function is in the range of 0 and 1, the input data is normalized to the smallest range, that the range of zero and one, using equation 16:

$$X_i' = \frac{X_i - X_{min}}{X_{max} - X_{min}} \qquad (16)$$

The goal of neural networks is to change the weight and bias matrix to reduce the error between the network output and the target values. In this paper, the normalized values of the inflow hydrograph are introduced as input, and the outflow hydrograph values are introduced as target data for simulation.

2.5. Data Availability

In this paper, the hydrograph from the data provided by Chow et al. (1988) has been routed using both neural networks and Muskingum's Method, employing the least squares error method. The results obtained from these approaches have been compared and presented alongside those from the study by Choudhury and Sankarasubramanian (2009). Additionally, the impact of the type of transfer function on the proposed neural network model has been examined. Table 1 and Figure 4 and 5 show data related to the hydrograph values of Chow et al. (1988) and the proposed model of Choudhury and Sankarasubramanian (2009).

Table 1 - Simulated data and values of the proposed model by Choudhury and Sankarasubramanian (2009)

Observational Hydrograph by Chow et al. (1988) (m^3/s)		The proposed model by Choudhury and Sankarasubramanian (2009) (m^3/s)			
		Flow model		Error model	
inflow	outflow	inflow	outflow	inflow	outflow
1.70	0.00	1.70	0.00	1.70	0.00
5.10	1.19	7.27	1.58	9.60	3.31
8.50	3.60	10.63	5.16	12.31	6.82
12.63	6.54	13.77	8.59	15.30	10.16
17.36	10.28	16.62	11.81	18.00	13.27
21.97	14.55	19.16	14.79	20.35	16.12
26.39	19.03	21.34	17.47	22.33	18.68
26.39	23.28	23.14	19.83	23.92	20.83
25.80	24.89	24.53	21.82	25.09	22.64
26.65	25.40	25.50	23.42	25.84	24.05
27.61	26.16	26.04	24.62	26.15	24.05
27.75	27.01	26.15	25.40	26.04	25.62
26.93	27.41	25.83	25.75	25.51	25.77
25.20	27.07	25.11	25.68	24.58	25.51
22.94	26.02	23.99	25.20	23.26	24.83
20.30	24.10	22.51	24.32	21.60	23.77
17.50	21.78	20.69	23.07	19.62	22.34
14.55	19.17	18.57	21.46	17.36	20.58
11.61	16.40	19.16	19.54	14.86	18.52
8.75	13.54	13.59	17.34	12.16	16.20
7.02	10.56	10.82	14.89	9.32	13.66
6.48	8.55	7.92	12.00	6.38	10.94
6.12	7.36	4.95	9.47	3.38	8.10
5.80	6.65	1.95	6.58	0.39	5.17

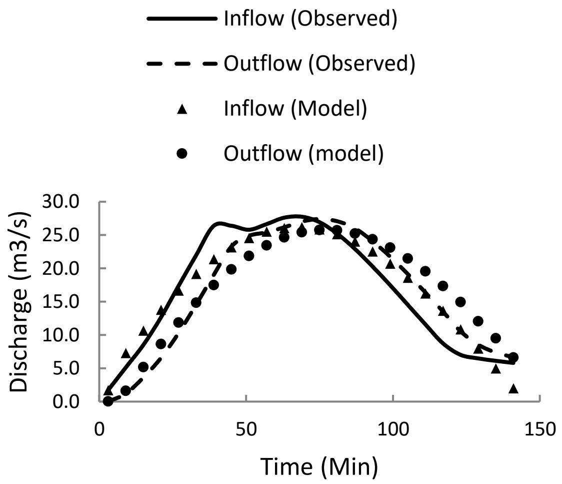

Figure 4. Observational inflow and outflow hydrograph of Chow et al. (1988)

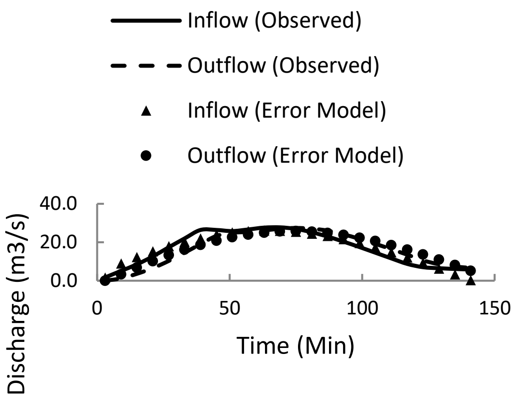

Figure 5. Flow model proposed by Choudhury and Sankarasubramanian (2009)

The reason for the difference in the data in Figure 4 and 5 are the use of different techniques in prediction. In this article, two methods includes the least squares error method, and neural networks have been used.

3. RESULTS

The networks utilized are Multi-Layer Perceptrons (MLP), which operate using a feed-forward back-propagation method. Research indicates that this type of network is effective for flood trend analysis and has been previously employed in related studies. Each network is structured with three layers: an input layer, an intermediate layer, and an output layer. The input layer consists of three neurons, while the output layer has one neuron that produces the output hydrograph. The performance function used is the Mean Squared Error (MSE), which evaluates the network's performance by calculating the average of the squared errors. The limitations on the number of layers and neurons in these networks are primarily due to the restricted amount of available data.

The performance of three trained networks is compared to the proposed model by Choudhury and Sankarasubramanian (2009). Figure 6 illustrates the effectiveness of the three transfer functions. Based on the mean absolute magnitude of the errors, we find that the tansig function has an error of 0.0359, the logsig function has an error of 0.0538, and the purelin function has an error of 0.0358. This indicates that both the tansig and purelin functions provide a better model. However, due to the superior prediction of peak discharge achieved by the tansig function, we accept it as the preferred model.

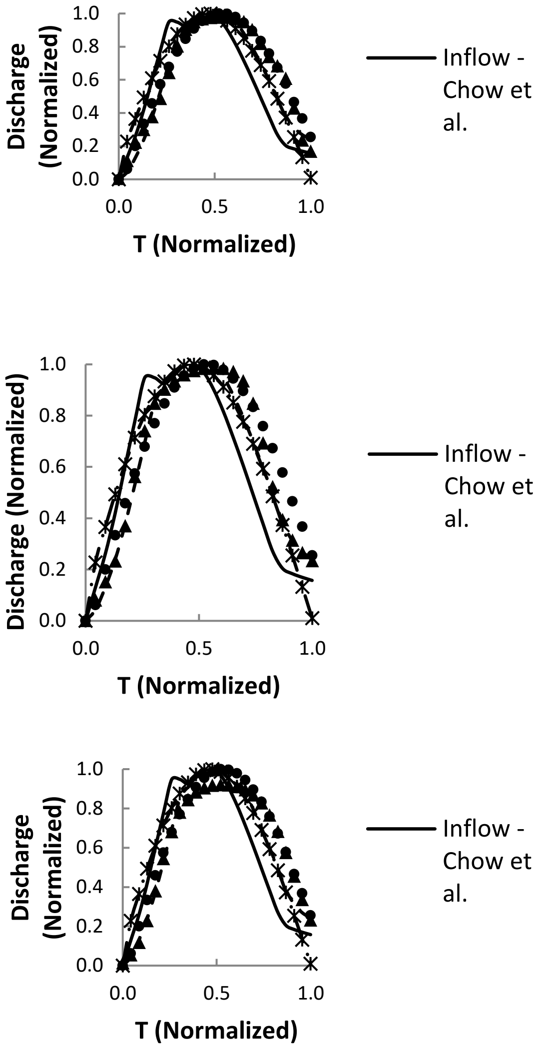

Figure 6. Comparison of the results of neural networks and the proposed model of Choudhury and Sankarasubramanian (2009) with different transfer functions including (a) tansig, (b)logsig, (c)purelin

In Figure 7, the proposed model is validated against the hydrograph developed by Choudhury and Sankarasubramanian (2009), referred to as the Error Model. The figure demonstrates that the proposed neural network model aligns well with the results of Choudhury and Sankarasubramanian's model.
Table 2 shows the mean absolute difference between the predictions of the neural network and the least squares method. From this data, we can conclude that while the neural network predictions are reasonably accurate, the least squares method has provided a relatively more precise value.

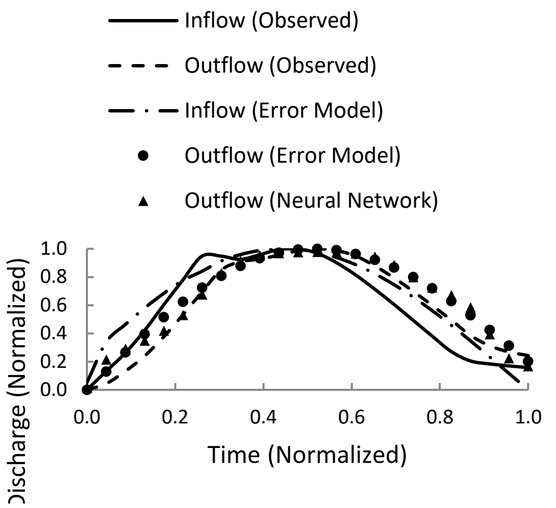

Figure 7. Comparison of results of neural networks and the proposed error model Choudhury and Sankarasubramanian (2009)

Figure 8 compares the results of the least squares method with those of the neural network.

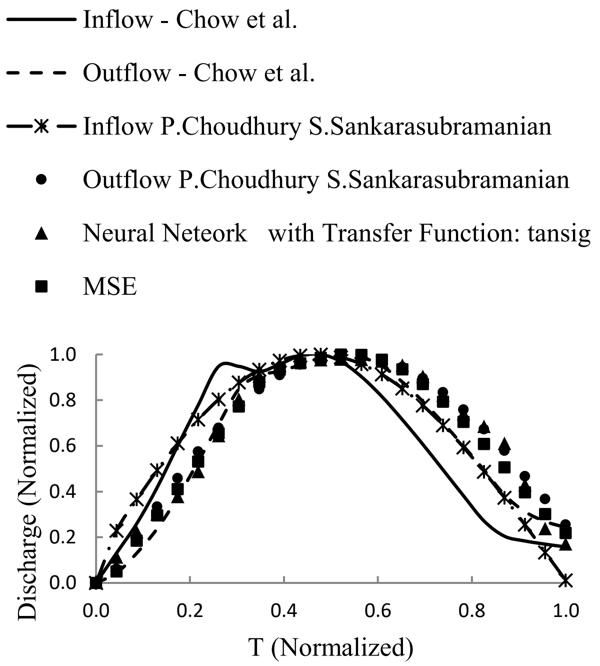

Figure 8. Comparison of results of neural networks, least squares error method, and the proposed model of Choudhury and Sankarasubramanian (2009)

Table 2 - Comparison of least squares error method and neural network

Method	neural networks	least squares error
RSME (%)	4.84	3.82

4. CONCLUSION

One of the most devastating disasters are Floods that can cause damage to ecosystems. Accurate simulation of floods are significantly important for flood control and the reduction of flood losses. There have been relatively limited studies in the field of reservoir routing using artificial intelligence models, and this is one of the innovations of the present article. This paper discusses flood routing using two methods: neural networks and the Muskingum method, which employs the least squares error approach. The models presented in this study are validated using the proposal by Choudhury and Sankarasubramanian (2009). Initially, the results of the neural network model using the transfer functions tansig, logsig, and purelin were examined. Ultimately, the proposed model was confirmed using the tansig function. The results were then compared to those obtained using the least squares error method. It was concluded that although both methods provide acceptable predictions, the least squares error method yields a relatively more accurate result.

References

[1] Agami, N., Atiya, A., Saleh, M., El-Shishiny, H. (2009). A neural network based dynamic forecasting model for Trend Impact Analysis. Technological Forecasting and Social Change, 76(7), 952-962.

[2] Al-Humoud, J. M., Esen, I. I. (2006). Approximate Methods For The Estimation Of Muskingum Flood Routing Parameters. Water Resources Management, 20(6), 979-990.

[3] Choudhury, P., Sankarasubramanian, A. (2009). River flood forecasting using complementary Muskingum rating equations. Journal of Hydrologic Engineering, 14(7), 745-751.

[4] Chow, V. T., Maidment, D. R., and Mays, L. W. (1988). Applied hydrology, International Ed., McGraw-Hill, New York.

[5] Chu, H.J., Chang, L.C. (2009). Applying particle swarm optimization to parameter estimation of the nonlinear Muskingum model. Journal of Hydrologic Engineering, 14(9), 1024-1027.

[6] Easa, S.M. (2013). Improved nonlinear Muskingum model with variable exponent parameter. Journal of Hydrologic Engineering, ASCE, 18(22), 1790-1794.

[7] Katipoğlu, O. M., Sarıgöl, M. (2023). Boosting flood routing prediction performance through a hybrid approach using empirical mode decomposition and neural networks: A case study of the Mera River in Ankara. Water Supply, 23(11), 4403-4415.

[8] Mirzazade, P. (2013). Investigation flood routing methods in river and reservoirs. M.Sc Thesis. Sistanand Baluchestan University. Civil college. Sistan and Baluchestan province. Iran. 86 (in Persian).

[9] Moghaddam, A., Behmanesh, J., Farsijani, A. (2016). Parameters estimation for the new four-parameter nonlinear Muskingum model using the particle swarm optimization. Water resources management, 30, 2143-2160.

[10] Mohan, S. (1997). Parameter estimation of nonlinear Muskingum models using genetic algorithm. Journal of hydraulic engineering, 123(2), 137-142.

[11] Niazkar, M., Afzali, S.H. (2014). Assessment of modified honey bee mating optimization for parameter estimation of nonlinear Muskingum models. Journal of Hydrologic Engineering, 20(4), p.04014055.

[12] Perumal, M. (1994). Hydrodynamic derivation of a variable parameter Muskingum method: 1. Theory and solution procedure. Hydrological sciences journal, 39(5), 431-442.

[13] Sarıgöl, M. (2024). Evaluating the Accuracy of Machine Learning, Deep Learning and Hybrid Algorithms for Flood Routing Calculations. Pure and Applied Geophysics, 181(12), 3485-3506